

Northumbria Research Link

Citation: Mistry, Kamlesh (2016) Intelligent facial expression recognition with unsupervised facial point detection and evolutionary feature optimization. Doctoral thesis, Northumbria University.

This version was downloaded from Northumbria Research Link:
<https://nrl.northumbria.ac.uk/id/eprint/36011/>

Northumbria University has developed Northumbria Research Link (NRL) to enable users to access the University's research output. Copyright © and moral rights for items on NRL are retained by the individual author(s) and/or other copyright owners. Single copies of full items can be reproduced, displayed or performed, and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided the authors, title and full bibliographic details are given, as well as a hyperlink and/or URL to the original metadata page. The content must not be changed in any way. Full items must not be sold commercially in any format or medium without formal permission of the copyright holder. The full policy is available online: <http://nrl.northumbria.ac.uk/policies.html>

Northumbria Research Link

Citation: Mistry, Kamlesh (2016) Intelligent facial expression recognition with unsupervised facial point detection and evolutionary feature optimization. Doctoral thesis, Northumbria University.

This version was downloaded from Northumbria Research Link:
<http://nrl.northumbria.ac.uk/id/eprint/36011/>

Northumbria University has developed Northumbria Research Link (NRL) to enable users to access the University's research output. Copyright © and moral rights for items on NRL are retained by the individual author(s) and/or other copyright owners. Single copies of full items can be reproduced, displayed or performed, and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided the authors, title and full bibliographic details are given, as well as a hyperlink and/or URL to the original metadata page. The content must not be changed in any way. Full items must not be sold commercially in any format or medium without formal permission of the copyright holder. The full policy is available online: <http://nrl.northumbria.ac.uk/policies.html>

**Intelligent Facial Expression
Recognition with Unsupervised
Facial Point Detection and
Evolutionary Feature Optimization**

K K Mistry

PhD

2016

**Intelligent Facial Expression
Recognition with Unsupervised
Facial Point Detection and
Evolutionary Feature Optimization**

Kamlesh Kakasaheb Mistry

A Thesis Submitted to
Northumbria University at Newcastle
for a Degree of Doctor of Philosophy

Department of Computer Science and
Digital Technologies

Faculty of Engineering and
Environment

August 2016

Acknowledgement

Many people have supported me during my PhD, and I would like to thank them all. Especially, I would like to thank my supervisor, Dr Li Zhang, for giving me top quality guidance and support during my PhD. She gave her quality time to shape my research ideas and paper development and proof-read my dissertation. Without her help, this dissertation would have never been possible. I would like to thank my family, especially my father and elder brother for their moral and financial support. I would also like to thank my lovely wife, Jyoti Mistry Jasekar. She always helped me to keep my mind calm and without her love I would have never gone this far. Also, I appreciate all the staff members who supported me and gave me guidance during my student life. I would also like to thank all of my fellow researchers from my Lab for being supportive and friendly. Finally, I would also like to thank the internal and external examiners for their time spent on reading the thesis and providing helpful suggestions.

Declaration

I declare that the work contained in this thesis has not been submitted for any other award and that it is all my work. I also confirm that this work fully acknowledges opinions, ideas and contributions from the work of others.

Any ethical clearance for the research presented in this thesis has been approved. Approval has been sought and granted by the Faculty Ethics Committee on 03/2016.

I declare that the Word Count of this thesis is 37656 words.

Name: Kamlesh Mistry

Signature:

Abstract

Facial expression is one of the effective channels to convey emotions and feelings. Many shape-based, appearance-based or hybrid methods for automatic facial expression recognition have been proposed. However, it is still a challenging task to identify emotions from facial images with scaling differences, pose variations, and occlusions. In addition, it is also difficult to identify significant discriminating facial features that could represent the characteristic of each expression because of the subtlety and variability of facial expressions. In order to deal with the above challenges, this research proposes two novel approaches: unsupervised facial point detection and texture-based facial expression recognition with feature optimisation.

First of all, unsupervised automatic facial point detection integrated with regression-based intensity estimation for facial Action Units (AUs) and emotion clustering is proposed to deal with challenges such as scaling differences, pose variations, and occlusions. The proposed facial point detector can detect 54 facial points in images of faces with occlusions, pose variations and scaling differences. We conduct AU intensity estimation respectively using support vector regression and neural networks for 18 selected AUs. FCM is also subsequently employed to recognise seven basic emotions as well as neutral expressions. It also shows great potential to deal with compound and newly arrived novel emotion class detection. The second proposed system focuses on a texture-based approach for facial expression recognition by proposing a novel variant of the local binary pattern for discriminative feature extraction and Particle Swarm Optimization (PSO)-based feature optimisation. Multiple classifiers are applied for recognising seven facial expressions.

Finally, evaluations are conducted to show the efficiency of the above two proposed systems. Evaluated using well-known facial databases: Helen, labelled faces in the wild, PUT, and CK+ the proposed unsupervised facial point detector outperforms other supervised landmark detection models dramatically and shows excellent robustness and capability in dealing with rotations, occlusions and illumination changes. Moreover, a comprehensive evaluation is also conducted for the proposed texture-based facial expression recognition with mGA-embedded PSO feature optimisation. Evaluated using the CK+ and MMI benchmark databases, the experimental results indicate that it outperforms other state-of-the-art metaheuristic search methods and facial emotion recognition research reported in the literature by a significant margin.

TABLE OF CONTENTS

| CHAPTER | PAGE |
|--|-----------|
| CHAPTER 1 INTRODUCTION..... | 1 |
| 1.1 CONTRIBUTIONS..... | 3 |
| 1.2 STRUCTURE OF THESIS..... | 6 |
| 1.3 PUBLICATIONS..... | 7 |
| CHAPTER 2 LITERATURE REVIEW | 8 |
| 2.1 FACIAL FEATURE EXTRACTION | 8 |
| 2.1.1 <i>Shape-based feature extraction.....</i> | <i>8</i> |
| 2.1.2 <i>Texture based feature extraction.....</i> | <i>12</i> |
| 2.2 FEATURE SELECTION TECHNIQUES | 14 |
| 2.2.1 <i>Feature selection using non-evolutionary algorithms.....</i> | <i>14</i> |
| 2.2.2 <i>Feature selection using evolutionary algorithms.....</i> | <i>16</i> |
| 2.3 ACTION UNIT INTENSITY ESTIMATION AND EMOTION RECOGNITION..... | 21 |
| 2.4 SUMMARY..... | 27 |
| CHAPTER 3 SUPERVISED FACIAL FEATURE EXTRACTION USING AAM IN COMBINATION WITH BRISK..... | 28 |
| 3.1 FEATURE EXTRACTION | 28 |
| 3.1.1 <i>Introduction to active appearance model.....</i> | <i>28</i> |
| 3.1.2 <i>Facial feature extraction using AAM.....</i> | <i>30</i> |
| 3.1.3 <i>Facial feature extraction using AAM and BRISK.....</i> | <i>34</i> |
| 3.2 FACIAL ACTION UNIT AND EMOTION DETECTION..... | 37 |
| 3.2.1 <i>Facial action unit intensity estimation.....</i> | <i>38</i> |
| 3.2.2 <i>Facial expression recognition.....</i> | <i>40</i> |
| 3.3 EVALUATION..... | 41 |
| 3.4 SUMMARY..... | 46 |

| | |
|---|-----------|
| CHAPTER 4 UNSUPERVISED FACIAL POINT DETECTION AND EMOTION RECOGNITION | 47 |
| 4.1 OVERVIEW OF THE PROPOSED METHODOLOGY | 47 |
| 4.2 UNSUPERVISED FACIAL POINT DETECTION..... | 51 |
| 4.2.1 Face and region of interest detection..... | 53 |
| 4.2.2 Gabor filtering based feature extraction..... | 54 |
| 4.2.3 Facial point detection using BRISK..... | 56 |
| 4.2.4 Facial point detection for occlusion using ICP and FCM | 60 |
| 4.3 AU INTENSITY ESTIMATION AND EMOTION CLUSTERING | 66 |
| 4.3.1 AU intensity estimation..... | 66 |
| 4.3.2 Facial emotion clustering | 70 |
| 4.4 EVALUATION..... | 77 |
| 4.4.1 Evaluation of facial landmark generation..... | 77 |
| 4.4.2 Evaluation of landmarks generation for images with occlusion..... | 82 |
| 4.4.3 Evaluation of landmarks generation for images with rotations..... | 85 |
| 4.4.4 Evaluation of AU intensity estimation and emotion recognition..... | 86 |
| 4.4 SUMMARY..... | 90 |
| CHAPTER 5 A MICRO-GA EMBEDDED PSO FEATURE SELECTION APPROACH TO INTELLIGENT FACIAL EMOTION RECOGNITION | 92 |
| 5.1 THE PROPOSED FACIAL EXPRESSION RECOGNITION SYSTEM | 92 |
| 5.1.1 Facial feature extraction using the proposed LBP model..... | 93 |
| 5.1.2 The proposed mGA-embedded particle swarm optimisation for feature selection..... | 95 |
| 5.1.2.1 Updating personal best and global best..... | 98 |
| 5.1.2.2 Constructing the secondary swarm embedded with the concept of mGA... | 101 |
| 5.1.2.3 In-depth local optimal search | 104 |
| 5.1.3 Emotion recognition..... | 105 |
| 5.2 EVALUATION..... | 107 |
| 5.2.1 Comparison of feature extraction algorithms..... | 107 |

| | |
|---|------------|
| 5.2.2 Comparison of feature selection algorithms..... | 109 |
| 5.2.3 Comparison of emotion recognition systems..... | 117 |
| 5.3 SUMMARY..... | 118 |
| CHAPTER 6 CONCLUSION AND FUTURE WORK | 119 |
| 6.1 SUMMARY OF CONTRIBUTIONS | 119 |
| 6.1.1 A supervised feature extraction model using AAM in combination with BRISK | 119 |
| 6.1.2 Unsupervised facial point detector for facial emotion recognition..... | 120 |
| 6.1.3 The texture based facial emotion recognition using hvnLBP feature extraction and mGA-embedded PSO feature selection..... | 121 |
| 6.2 LIMITATIONS OF THIS APPROACH..... | 121 |
| 6.3 FUTURE WORK..... | 122 |
| 6.3.1 Combining bodily gestures with facial expressions to enhance emotion recognition performance..... | 122 |
| 6.3.2 Exploring micro-emotions..... | 122 |
| 6.3.3 Exploring various classifiers for emotion recognition | 123 |
| 6.3.4 Various applications for the proposed models..... | 123 |
| REFERENCES LIST | 125 |

List of Abbreviations

| | |
|--------|--|
| AAM | Active Appearance Model |
| ABC | Artificial Bee Colony |
| ACS | Adaptive Cuckoo Search |
| ANN | Approximate Nearest Neighbour |
| ASM | Active Shape Model |
| AUDN | Action Unit inspired Deep Networks |
| AUs | Action Units |
| BBPSO | Binary Bare-Bone Particle Swarm Optimisation |
| BBPSO | Binary Bare-bone Particle Swarm Optimisation |
| BPSO | Bare Bone Particle Swarm Optimisation |
| BRIEF | Binary Robust Independent Elementary Features |
| BRISK | Binary Robust Invariant Scalable Keypoints |
| CBPSO | Chaotic Binary Particle Swarm Optimisation |
| CCA | Curvilinear Component Analysis |
| CCM | Cross-Correlation Model |
| CFA | Chaotic Firefly Algorithm |
| CK+ | Extended Cohn-Kanade |
| CLBP | Completed Local Binary Pattern |
| CLM | Constrained Local Model |
| CMDPSO | Crossover Mutation Dominance Particle Swarm Optimisation |
| CS | Cuckoo Search |

| | |
|----------|---|
| CS-LBP | Centre-Symmetric Local Binary Pattern |
| DE | Differential Evolution |
| DFA | Discrete Firefly Algorithm |
| DisABC | Discrete Artificial Bee Colony |
| DLBP | Dominant Local Binary Pattern |
| DS-GPLVM | Discriminative Shared Gaussian Process Latent Variable Model |
| EC | Evolutionary Computational |
| EEG | Electroencephalogram |
| ELPSO | Enhanced Leader Particle Swarm Optimisation |
| EWCCM | Error Weighted Cross-Correlation |
| FA | Firefly Algorithm |
| FACS | Facial Action Coding System |
| FAST | Features from Accelerated Segment Test |
| FCM | Fuzzy C-Means |
| FDP | Facial Deformation Parameters |
| FLD | Fisher Linear Discriminant |
| GA | Genetic Algorithm |
| GBDE | Gaussian Bare-bone Differential Evolution |
| GMM | Gaussian Mixture Model |
| GN-DPM | Gauss-Newton Deformable Part Models |
| gViSOM | Growing variant of Visualization Induced Self-organizing Maps |
| HCI | Human Computer Interaction |
| HEPSO | High Exploration Particle Swarm Optimisation |

| | |
|--------|--|
| HMM | Hidden Markov Model |
| HS | Harmony Search |
| hvnLBP | Horizontal Vertical Neighborhood Local Binary Pattern |
| IC | Inverse Compositional |
| ICP | Iterative Closest Point |
| KPCA | Kernel Principle Component Analysis |
| LBP | Local Binary Pattern |
| LDP | Local Derivative Pattern |
| LFW | Labelled Faces in Wild |
| LGBP | Local Gabor Binary Pattern |
| LLE | Local Linear Embedding |
| LPP | Locality Preserving Projections |
| LPQ | Local Phase Quantization |
| MAP | Micro-Action-Pattern |
| MFA | Memetic Firefly Algorithm |
| MFOPSO | Multimodal Function Optimisation Particle Swarm Optimisation |
| mGA | Micro Genetic Algorithm |
| MI | Mutual Information |
| mRMR | Minimum Redundancy Maximum Relevance |
| MSE | Mean Square Error |
| NN | Neural Network |
| NSGAI | Non-dominated Sorting Genetic Algorithm II |
| NSPSO | Non-dominated Particle Swarm Optimisation |

| | |
|--------|--|
| PCA | Principle Components Analysis |
| PSO | Particle Swarm Optimisation |
| RBF | Radial Basis Function |
| ROI | Region of Interest |
| RVR | Relevance Vector Regression |
| SA | Simulated Annealing |
| SOM | Self-organizing Maps |
| SURF | Speeded-up Robust Features |
| SVM | Support Vector Machine |
| SVR | Support Vector Regression |
| ThBPSO | Threshold-based Binary Particle Swarm Optimisation |
| ViSOM | Visualisation Induced Self-organizing Maps |

Chapter 1 Introduction

Automatic facial expression recognition has been intensively studied in computer vision and has been widely used in variety of applications such as personalized healthcare (Lucey et al., 2009), human-computer interaction (HCI) (Bartlett et al., 2003), drowsiness detection (Vural et al., 2008), and humanoid robots (Mistry et al., 2015; Zhang et al., 2015). Ekman and Friesen (1976) conducted thorough research on facial expressions and concluded a set of universally recognisable six basic emotions: happiness, anger, sadness, surprise, disgust and fear. Facial Action Coding System (FACS) (Ekman et al., 2002) was introduced for coding facial muscle movements and extract the detailed information from facial expressions.

Moreover, FACS has been used as an intermediate channel to bridge raw motion-based facial representations with emotional facial behaviour recognition in many applications (Zeng et al., 2009). FACS provides an objective approach to describe the truth of human behaviour and is closely related to physical indicators of emotional facial expressions. It employs 32 action units (AUs), which represent the muscular activities to describe and identify the intensity of facial expressions. It also provides a versatile method to describe a broad range of facial behaviours, e.g. facial punctuators in conversation and emotional facial expressions (Ekman & Friesen, 1976; Ekman et al., 2002). FACS is also capable of describing emotion intensities and compound emotions and distinguishing fake from real emotional expressions. Thus many computational facial emotion recognition studies employed FACS and AUs (Zeng et al., 2009).

According to Ekman et al. (2002), the intensity of an AU can be scored on a five-point ordinal level from A to E. Level A refers to a trace of an action. Level B indicates slight evidence. Level C describes the obvious or marked evidence. Level D represents severe or extreme actions with Level E indicating maximum evidence. Each intensity level refers to a range of appearance changes. Despite intensive studies of facial AU detection, automatic AU intensity measurement still poses significant challenges to automated recognition systems since the differences between some AUs' intensity levels could be

subtle and subjective during emotional or conversational behaviour, and the physical cues of one AU might vary greatly when it occurs simultaneously with other AUs.

Efficient facial representations play a vital role in achieving robust facial emotion classification. First of all, it is a challenging task to identify emotions from facial images with pose variations, illumination changes, occlusions and background clutter. Many supervised parametric and mode specific shape based feature extraction approaches have been proposed to detect facial landmarks from real-life images to benefit subsequent automatic facial behaviour perception to address the above issues. However, many of the above applications found it difficult to balance well between high-quality feature extraction and low computational requirements, which is essential for real-time applications. Although various approaches have been used to improve their robustness and efficiency, they usually require intensive computational convergence time. Thus, other computational-wise optimal unsupervised methods for automatic face analysis are necessary to deal with challenging real-life (spontaneous) facial emotion recognition tasks.

In comparison with shape features, texture features can provide more discriminative and emotion-specific information. However, they can also be affected by illumination changes, scaling and rotation variations, which eventually leads to poor classification accuracy. Various texture feature extraction algorithms have been proposed in the literature to extract discriminative and high-quality features. Although such methods achieved impressive performance, the extracted features tend to be high dimensional and makes texture-based methods computationally expensive and less applicable in real-time situations. Although various algorithms have also been proposed for facial feature selection (Lajevardi & Hussain, 2012), it is still a challenging task to extract the significant discriminative facial feature subsets that represent the characteristics of each emotion because of the subtlety and variations of facial expressions.

To address the above challenges, we offer three automatic facial expression recognition systems. Firstly, we propose a supervised facial feature extraction model by integrating Active Appearance Model (AAM) (Cootes et al., 2001) with Binary Robust Invariant Scalable Keypoints (BRISK) (Leutenegger et al., 2011) to improve the geometric facial fitting and tracking and to extract shape and texture features. The facial feature

extraction model using AAM with BRISK shows high efficiency in facial landmark detection to inform subsequent facial expression recognition (Mistry et al., 2015). To reduce computational cost and address limitations of supervised facial feature extraction models, secondly, we propose a novel unsupervised facial point detector to extract facial features from images with illumination variations, rotation changes, and facial occlusions, more efficiently. This unsupervised facial extraction model combines a series of processes such as Gabor filter, BRISK, an Iterative Closest Point (ICP) algorithm and fuzzy c-means (FCM) clustering for efficient facial landmark detection and feature extraction, which outperforms supervised models greatly in diverse facial expression recognition tasks. Finally, we propose a texture-based facial expression recognition system using a novel variant of Local Binary Patterns (LBP) operator for discriminative feature extraction and Particle Swarm Optimization (PSO)-based feature optimisation. Multiple classifiers are employed for recognising seven facial expressions. Evaluated with diverse databases, it outperforms other optimisation methods and existing state-of-the-art facial expression recognition research significantly.

1.1 Contributions

The major contributions of this thesis are categorised into three aspects as follows:

1. First of all, we present a supervised facial feature extraction model using AAM in combination with BRISK.
 - In this research, we present a hybrid approach with the consideration of shape based features for facial action and emotion recognition. First of all, we develop a rotation-invariant novel feature extraction model to extract shape features from facial images. This feature extraction method integrates AAM with a novel feature descriptor, BRISK to deal with rotations, illumination changes, and head movements. The resulting features from the model are subsequently used for facial action and emotion recognition. This feature extraction model can derive 68 two-dimensional motion-based landmark points from a raw facial image. The generated shape information by AAM with BRISK fitting provides comparatively more accurate and efficient face representations owing to BRISK's robustness to

rotations. Especially, the model can perform such effective feature extraction from images with rotations and pose variations without prior related specialised knowledge required. This system has been reported in Chapter 3. Related research is presented in Mistry et al. (2015).

- The facial features extracted from the methods mentioned above are further used as input to the neural network (NN) classifier to detect 18 AUs. These detected AUs are used as input to the emotion classifier in order to detect seven basic emotions. The overall system is evaluated using the well-known Extended Cohn-Kanade (CK+) (Kanade et al., 2000; Lucey et al., 2010) facial expression database.
2. The second major contribution of this research is an unsupervised facial point detector to deal with emotion detection from images with pose variations, illumination changes, occlusions and background noise.
 - A cost-effective Optimal unsupervised learning scheme for facial point detection is implemented by incorporating a 2D Gabor filter, BRISK, the ICP algorithm and FCM.
 - The unsupervised facial point detector applies ICP first to recover neutral landmark points for an occluded facial region. Then FCM has applied to further inference the shape of the occluded element by taking the prior knowledge of the attributes of the non-occluded facial regions into account. After we have applied FCM to obtain the shape cluster of the occluded facial element, we select the top five image outputs with the highest correlations to the test image in the cluster and average them to reconstruct the best suitable geometry for the occluded facial element. The reconstructed set of landmarks with shape information embedded for the occluded facial region is then used to adjust the neutral landmarks generated by ICP.
 - This background robust facial feature point detector can detect 54 facial points for images or real subjects from challenging real-time human-computer interaction by a high efficiency to fulfil computational constraints. This system has been reported in Chapter 4 and presented in Zhang et al. (2015).
 3. The third major contribution of this research is a texture based facial emotion recognition system using a new modified LBP feature extraction algorithm and a Micro

Genetic Algorithm (mGA) embedded PSO feature optimisation. The system addresses the challenges such as illumination changes, rotation variations, and high dimensionality of features.

- This texture-based facial expression recognition, first of all, presents a modified LBP operator for feature extraction that conducts horizontal and vertical neighbourhood pixel comparison, to overcome the drawbacks of original LBP by retrieving the missing contrast information embedded in the neighbourhood to generate the initial discriminative facial representation.
- A novel mGA-embedded PSO algorithm is implemented for feature optimisation, to mitigate the premature convergence and local optimum problems of conventional PSO. It provides great flexibility to allow the feature selection process to separate not only facial features into specific areas for in-depth local search but also combine facial features for overall global search.
- The mGA-embedded PSO algorithm includes a new velocity updating strategy by employing the personal average experience to generate the individual best, *pbest*, and Gaussian mutation to produce the global best, *gbest*, to increase swarm diversity. The proposed algorithm also applies the diversity maintenance strategy of mGA to keep the original swarm in a non-replaceable memory (Coello & Pulido, 2001), which remains intact during the lifetime of the algorithm, to reduce the probability of premature convergence.
- In order to speed up evolution for convergence, the small population size concept of mGA is used to generate a secondary swarm with five particles. The secondary swarm consists of the swarm leader and four follower particles from the non-replaceable memory with the lowest or highest correlation with the leader to increase local exploitation and global exploration. These local and global search mechanisms work in a collaborative manner to guide the search towards global optima.
- A sub-dimension based search strategy is also conducted, to identify optimal features for each facial region. The overall proposed system is evaluated with CK+

and MMI (Pantic et al., 2005) databases. The proposed system has been reported in Chapter 5 and presented in Mistry et al. (2016).

1.2 Structure of Thesis

The structure of the rest of the thesis is introduced in the following.

Chapter 2 presents a detailed literature review of related research. Specifically, it provides related work discussion on facial feature extraction, feature optimisation and facial emotion classification.

Chapter 3 presents the supervised facial feature extraction model using AAM in combination with BRISK. Detail evaluation results using diverse databases are also provided. Comparison with related research is also conducted to show the system's efficiency.

Chapter 4 presents the unsupervised facial point detector for facial feature extraction and expression recognition. It includes the introduction of each key technique applied such as 2D Gabor Filter, a feature descriptor BRISK, and ICP and FCM. The unsupervised facial point detector can deal with rotations, illumination changes and occlusions. Then we present the system evaluation results for AU intensity estimation and emotion recognition using this unsupervised facial point detector.

Chapter 5 presents a texture-based facial expression recognition system with a novel horizontal and vertical neighbourhood LBP based feature extraction and mGA-embedded PSO feature optimisation. It also presents various emotion classification algorithms such as artificial NN, support vector machines (SVMs) and ensemble classifiers. Then detailed evaluation results in comparison with related optimisation methods and related facial expression recognition research are also discussed.

Chapter 6 summarises the major contributions of this research and identifies future directions.

1.3 Publications

- Zhang, L., Mistry, K., Hossain, A.M., Shape and Texture based Facial Action and Emotion Recognition, in: 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), May 2014.
- Mistry, K., Zhang, L., Neoh, S.C., Jiang, M. Intelligent Appearance and shape based facial emotion recognition for a humanoid robot, SKIMA 2014.
- Mistry, K., Zhang, L., Intelligent Facial Expression Recognition with Adaptive Feature Extraction for a Humanoid Robot, in: International Joint Conference on Neural Networks (IJCNN), July 2015.
- Neoh, S.C., Zhang, L., Mistry, K., Hossain, M.A., Lim, C.P., Aslam, N., Kinghorn, P. Intelligent facial emotion recognition using a layered encoding cascade optimization model, Applied Soft Computing 34 (2015) 72-93, doi: 10.1016/j.asoc.2015.05.006.
- Zhang, L., Mistry, K., Jiang, M., Neoh, S.C., Hossain, M.A., Adaptive facial point detection and emotion recognition for a humanoid robot, Computer Vision and Image Understanding 140 (2015) 93-114.
- Mistry, K., Zhang, L., Neoh, S.C., Lim, C.P., Fielding, B., A micro-GA Embedded PSO Feature Selection Approach to Intelligent Facial Emotion Recognition, IEEE transaction on Cybernetics, 2016.

Chapter 2 Literature Review

This Chapter presents the literature review of state-of-the-art research on automatic facial feature extraction, feature optimisation, AU intensity estimation and facial emotion recognition.

2.1 Facial Feature Extraction

Face analysis has become a very popular research topic due to its wide range of applications such as face recognition, facial emotion recognition and gaze detection. This section discusses several key applications with impressive performances for facial feature extraction. This section is further categorised into two parts: shape-based and texture-based feature extraction.

2.1.1 Shape-based feature extraction

This section discusses related techniques and applications for shape-based feature extraction. Vukadinovic and Pantic (2005) used Gabor feature based boosted classifiers for the detection of 20 facial points. Their system first divided a detected facial region into 20 regions of interest and applied GentleBoost templates integrating grey level intensities and Gabor wavelet features for facial point detection. However, their work had not been developed to deal with various head rotations and occlusions. Senechal et al. (2011) presented a facial feature point detection system to automatically track 18 facial landmarks (i.e. four per eye, three per brow and four for the mouth). The tracking system was developed based on improved multi-kernel learning. It combined facial feature points matching between consecutive images with a prior knowledge of facial landmarks. This point matching procedure employed both static (i.e. patches) and dynamic features (i.e. the correlations between the current pixel patch of the region of interest (ROI) and those extracted from the previous image sequence) as inputs to a multi-kernel SVM. The output of SVM provided a confidence index of being the searched landmark or not for each candidate pixel. Their system showed efficient in dealing with real-time facial feature point detection.

Facial shape and appearance modelling techniques have also been intensively studied. These include supervised models such as AAM, Active Shape Model (ASM), Constrained Local Model (CLM) and regression-based algorithms. They have been widely used for geometric and texture facial feature extraction. We discuss these models and their related applications in the following.

AAM was initially proposed by Cootes et al. (1998). It can generate both shape and texture representations of deformable objects and showed to be very efficient and flexible for objects tracking in many applications (Cootes et al., 1998). According to Matthews and Baker (2004), there are two types of linear AAMs: independent and combined AAMs. Independent AAMs perform linear modelling of shape and appearance of deformable objects separately, while combined AAMs employ a single set of linear parameters to describe shape and appearance. The training of an AAM requires a set of images and coordinates of landmarks associated with these images as inputs.

The task of AAM fitting is to search for the set of shape and appearance parameters which offer the best fitting between the trained model and the given input image. Many model fitting algorithms and strategies have been proposed in the past to reduce the error between the given input image and the calculated model. One comparatively more efficient and computational affordable approach is to update the model with an incremental warp using the inverse compositional algorithm. This algorithm has been regarded as the more efficient warp compositional algorithm for AAM fitting (Matthews & Baker, 2004). However, research showed that the trained texture models of AAM were often not robust enough to reconstruct generic faces. Also, for real-time fitting, such reconstructions errors dominated the alignment errors, which tended to lead to poor performances (Jaiswal et al., 2013). There were also significant efforts made further to improve the robustness, efficiency and discriminative abilities of AAM. For example, some researchers have made explorations to build AAM in a 3D space to model scaling, translations and rotations more efficiently (Gao et al., 2010). Sung and Kim (2007) also proposed a background robust AAM with the assistance of Active Contour Model. In their approach, the active contour technique was used to detect the face boundary (the object foreground) from a cluttered image background first. AAM fitting was then applied to the selected foreground of the

object. The discrimination of AAM depends heavily on the accuracy of the fitting and efficient AAM based face tracking also has great potential to deal with head movement and pose variations. Gross et al. (2004) explored adaptive fitting of AAM and proposed a new fitting model robust enough to deal with image occlusions. To improve the AAM fitting, we have presented a novel system which employs BRISK algorithm. The detailed discussion on this model has been given in Chapter 3.

CLM was proposed by Cristinacce and Cootes (2006). This model employs a class of methods to locate sets of points on a target image, which is constrained by a statistical shape model. Although the fitting accuracy still needed improvements, compared to AAM, CLM offers better real-time efficiency and robustness. Various strategies and optimisation methods were also employed to improve the computational efficiency and accuracy for CLM. For example, Asthana et al. (2013) proposed a robust discriminative response map fitting method for face fitting. It integrated a discriminative regression based approach with CLM. Compared to AAM, their approach was able to use a small effective set of parameters to record and reconstruct response maps. Their method also proved to be able to perform real-time facial point detection efficiently. Experiments conducted with several facial image databases proved the robustness and efficiency of the proposed method for generic face fitting tasks. Gauss-Newton Deformable Part Models (GN-DPMs), for face alignment, have been suggested by Tzimiropoulos and Pantic (2014). Their model employed Gauss-Newton optimisation to minimise a joint cost function of the shape and appearance to generate a joint translational motion model. It jointly optimised a part-based linear generative appearance model and a global shape model to overcome the limitations of the traditional DPMs which were based on independent fixed part templates and tended to show detection ambiguity. Their model achieved a computational reduction in the training stage with the proposed cost function further evaluated during fitting. Their experiments indicated that GN-DPMs outperformed other models proposed in prior work.

There are also other regression-based methods for facial feature detection. Martinez et al. (2013) developed a novel algorithm for face fitting and facial point detection for frontal and near-frontal images. The algorithm combined a low computation cost regression-based approach with the robustness of exhaustive search based face shape

model. It relied on the regression method to conduct a quality measure of each prediction and employed the exhaustive search based shape model to provide correction to the sampling region.

Head rotation and pose variation estimation is a challenging topic and essential to achieving advanced performance for facial feature extraction and emotion recognition. Zhang et al. (2013) proposed a sparse representation based latent space model to deal with face recognition tasks with large pose variations. Their method employed the sparsity property of a face image over its corresponding view-dictionary to deal with pose difference between gallery and probe faces. Orozco et al. (2013) produced a pose-robust online appearance based facial feature tracker for the simultaneous tracking of head pose, lips, eyebrows, eyelid and irises from video sequences. This tracking model was able to track both face pose and eye gaze simultaneously in comparison to other tracking techniques. It also integrated with a dynamic learning mechanism to deal with appearance changes of the tracked object without the requirement of prior training. A very robust facial analysis framework was employed in their work by incorporating three of such facial feature trackers and the application of a Levenberg-Marquardt optimisation algorithm to deal with pose variations and occlusions.

Furthermore, since facial expressions relate to upper, middle, and lower regions of the face with similar facial muscle contractions in each region, an occluded facial component (e.g. eyes or mouth) of a facial expression could be predicted by using the statistical dependency among features from paired facial components on non-occluded regions. In Lin et al. (2013) and Wu et al. (2014), an Error Weighted Cross-Correlation Model (EWCCM) was proposed to generate a set of Gaussian Mixture Model (GMM) based Cross-Correlation Model (CCM) predictors to predict AU or AU combination of the occluded facial component based on the facial deformation parameters (FDPs) extracted from the paired facial elements of the non-occluded regions. Within the EWCCM model, a Bayesian classifier weighting scheme was employed to integrate all the GMM-based CCM predictors, each for one paired facial component, to give the final decision on the predicted AU or AUs. The EWCCM and Bayesian classifier weighting scheme based prediction were able to achieve mediocre accuracy (Lin et al., 2013) and effectively

reconstruct the geometric facial features of the occluded region in facial emotion expressions (Wu et al., 2014).

2.1.2 Texture based feature extraction

The literature shows that the widely used texture feature extraction algorithms are Principal Component Analysis (PCA), LBP, and Gabor filter. However, among all other feature extraction algorithms, LBP algorithm is more popular due to its robustness to illumination changes and computational simplicity (Shan et al., 2009). However, LBP features fail to deal with rotation variations. A number of LBP variants are proposed to increase its robustness to the rotation and discriminative power. As an example, dominant LBP (DLBP) can retrieve the most frequently occurred patterns of LBP to improve its texture descriptive capability. According to (Liao et al., 2009), uniform patterns in LBP can lead to a loss of information with respect to complex shapes despite their effectiveness in capturing fundamental patterns in an input image. Therefore, instead of purely using uniform patterns, DLBP calculates the occurrence frequencies of all the patterns extracted by LBP. These patterns are subsequently ranked based on the occurrence frequencies to enable the extraction of dominating patterns in texture images.

Local Gabor Binary Patterns (LGBP) is used as a novel face representation approach by combining Gabor filter with LBP (Zhang et al., 2005). This feature extraction method applies Gabor filters on an input image first to produce Gabor Magnitude Pictures and then an LBP operator is applied. Therefore, LGBP has excellent representation and discriminating power of the spatial information of the face. However, LGBP features still suffer from high dimensionality. Completed LBP (CLBP) (Guo et al., 2010) employs three key components, i.e. CLBP-Centre, CLBP-Sign, and CLBP-Magnitude, to extract the image's local grey level, the sign and magnitude features of local difference, respectively. The final CLBP histogram is formed by fusing these three components. In comparison with LBP which only considers the sign element, CLBP takes the magnitude component and intensity of the central pixel into account for formulating the additional discriminative power. It produces superior texture classification accuracy than those from other state-of-the-art LBP algorithms. Centre-symmetric LBP (CS-LBP) (Heikkilä et al., 2009) aims to

solve the lengthy histogram problem of LBP. In order to produce more compact binary patterns, CS-LBP purely employs the center-symmetric pairs of pixels for comparison. Therefore, compared with LBP, it enables a significant reduction in dimensionality while capturing better gradient information.

Local derivative pattern (LDP) (Zhang et al., 2010) is a high-order local pattern descriptor, which encodes directional pattern features based on local derivative variations. In comparison with LBP (as a non-directional first-order local pattern operator), LDP encodes more detailed discriminative information by calculating higher-order directional derivatives. It effectively extracts spatial relationships in a local region. LBP, on the other hand, only defines the relationships between the central point and its neighbours. In LDP, the first-order derivatives from four different directions, i.e. 0° , 45° , 90° , and 135° , are calculated. A set of 16 spatial relationship templates is defined for derivative direction comparisons with each template assigned a value of 0 or 1 based on whether it is a ‘monotonically increasing/decreasing’ or a ‘turning point’ pattern. The four first-order derivatives are then concatenated to form the second-order LDP. The n^{th} -order LDP, therefore, encodes the $(n-1)^{\text{th}}$ -order derivative direction variations. Higher-order LDP possesses superior capabilities in providing detailed discriminative features, but at the cost of an increasing level of noise. Another novel texture descriptor, local phase quantization (LPQ) (Ojansivu & Heikkilä, 2008) deals with image blurring based on the quantized phase of the discrete Fourier transform computed in local neighbourhoods. The LPQ operator is tolerant to centrally symmetric blur including motion, out of focus, and atmospheric turbulence blur. It is developed based on the blur invariance characteristics of the Fourier phase spectrum. In LPQ, four Fourier coefficients are used to sample the phase component of the frequency at four discrete points for each individual pixel position. The resulting vector is then further processed by separating each value into the real and imaginary parts to generate an eight-dimensional vector. De-correlation is also conducted using a whitening transform to ensure the statistical independence of the samples. A simple scalar quantizer is subsequently used to obtain the 8-bit binary code for each pixel position representing a blur insensitive, Fourier phase information of the pixel location. These codes are then

converted into a histogram for image classification. Overall, LPQ is superior to LBP and Gabor filter bank-based methods in dealing with image blurring.

2.2 Feature Selection Techniques

Feature selection techniques are usually applied to deal with feature dimensionality reduction. Such algorithms can be grouped into two categories, i.e. non-evolutionary and evolutionary algorithms.

2.2.1 Feature selection using non-evolutionary algorithms

Many feature dimensionality reduction techniques have been proposed in the literature. Among them, PCA has been widely used for feature reduction in face recognition research for decades (Liu & Wechsler, 2000; Gosavi & Khot, 2013; Kaur et al., 2010). According to Swets and Weng (1996), PCA derives the most expressive features, but may not embed sufficient discriminating power. However, the empirical finding shows that PCA is unable to capture the non-linear structures of the data which makes it less efficient for complex data sets. In order to extend the PCA to deal with non-linearity, Scholköpfung et al. (1998) proposed a Kernel PCA (KPCA). The KPCA projects the data into a higher-dimensional feature space using an imaginary non-linear function and performs a standard PCA via a kernel function.

In addition to PCA, Fisher Linear Discriminant (FLD) is another commonly used feature reduction technique which is claimed to provide comparatively more class separability by maximizing the mean between classes and minimizing the variation within a class (Gu et al., 2012; Shah, 2014; Chavan & Kulkarni, 2013). Thus FLD projects the most discriminative features for class distinction. However, it requires a broad coverage of face/class variations at the training stage in order to yield a more superior recognition performance.

To solve the dimensionality reduction problem, Tenenbaum et al. (2000) proposed the Isomap algorithm. In order to learn the essential global geometry, the Isomap algorithm first constructs a neighbourhood graph, then computes the geodesic distances between all pairs of data points and finally applies multidimensional scaling to the grammian matrix.

Another dimensionality reduction method, Local Linear Embedding (LLE) was proposed by Roweis and Saul (2000) to map the high-dimensional nonlinear data onto a single global coordinate system of lower dimensional subspace. The LLE exploits the local symmetries of linear reconstructions to learn the global structures of nonlinear variations. He and Niyogi (2003) proposed a locality preserving projections (LPP) algorithm which generates the linear projective maps by solving a variation problem. In other words, LPP computes linear projection maps to optimally preserve the neighbourhood structure of the data set. The LPP algorithm was further extended to Orthogonal LPP by Cai et al. (2006), to obtain the most discriminative mapping.

A curvilinear component analysis (CCA) algorithm was proposed by Demartines and Herault (1997) which is also based on self-organizing neural networks to perform vector quantization and nonlinear projection. The CCA algorithm preserves local distance relationships to detect the essential geometric properties of the data. A self-organizing map (SOM) algorithm was proposed by Kohonen (1997) to solve the dimensionality reduction problem. The SOM algorithm uses unsupervised learning to generate a low-dimensional and discriminative representation of maps. The SOM can identify the following relationships of the data but is unable to replicate quantitative distances between the data points in the reduced space. Yin (2002) proposed a visualisation induced SOM (ViSOM), which preserves the local distance maps by regularising the inter-neuron distances within a neighbourhood. The empirical findings showed that the SOM based algorithms (e.g. ViSOM) have convergence problems, i.e. a SOM with a prefixed size cannot converge efficiently to highly nonlinear manifolds (Yin, 2008). To solve the above convergence problem, Yin (2008) proposed a growing variant of ViSOM (gViSOM) algorithm which implements growing structures such as entire rows or columns or incrementing polygons. The gViSOM produces metric scaling manifolds and extracts extremely nonlinear manifolds.

Moreover, the literature review shows that, branch and bound, sequential selection, mutual information (MI), and Minimum Redundancy Maximum Relevance (mRMR), have been employed for optimized feature selection (Ravindran et al., 2014; Zeng et al., 2014; Mohemmed et al., 2009; Ajit Krisshna et al., 2014). The branch and bound algorithm adopt

monotonicity assumptions when searching for the optimal feature subset, whereas sequential selection that adds or removes one feature at a time is computationally expensive (Ravindran et al., 2014). On the other hand, MI (Zeng et al., 2014) is an information theoretic feature selection method that is not limited to linear dependencies and can maximise information in class. mRMR (Zeng et al., 2014) was proposed to improve the performance of MI further. However, mRMR is limited to increment search schemes to select one feature at a time without considering interactions between groups of features (Zeng et al., 2014).

2.2.2 Feature selection using evolutionary algorithms

In comparison with non-evolutionary feature selection algorithms, evolutionary computational (EC) algorithms prove to be more efficient in global search capabilities and have been widely accepted as effective techniques for feature selection (Xue et al., 2013). Some popular EC algorithms include Genetic Algorithm (GA), Cuckoo Search (CS), PSO, firefly algorithm (FA) and Artificial Bee Colony (ABC).

Kirkpatrick et al. (1983) proposed simulated annealing (SA), which is a trajectory based random search technique for solving the optimisation problem. The annealing characteristics in metal processing inspired the concept of SA algorithm. SA first produces the neighbour solutions of the current solution using random walk strategies and compares the neighbour solutions with the current solution. If the current solution is worse than the neighbour solution, then it accepts the neighbour solution. However, if the neighbour solution is worse than the current solution, then it checks the probability rule and accepts the neighbour solution based on the probability rule. The probability rules are determined by temperature which decreases during the execution of the SA algorithm. The search strategy of SA shows the ability to reduce the risk of getting trapped in local minima by increasing global exploration. Mohammad and Salwani (2015) combined SA algorithm with FA to improve the classification accuracy of the probabilistic neural networks. Loderer et al. (2015) applied SA algorithm on top of LBP to select the most important facial features and improve the facial emotion recognition accuracy.

A nature inspired metaheuristic algorithm, named as GA was proposed by Holland (1975) to deal with complex problems and parallelism. GA follows the natural selection process based on Charles Darwin's biological evolution theory. GA depends on three key operators such as crossover, mutation and selection. In GA the evolution starts with a randomly generated population and in each iteration GA evaluates the fitness of each individual in the population. GA selects the fittest individuals from the current population and applies mutation to form a new generation. It continues until the population evolves to contain an acceptable solution. The literature also shows that GA becomes one of the most popular evolutionary algorithms regarding a wide range of applications. The GA is widely used for solving high dimensionality and feature selection problems. In order to improve the exploration of the search space, many hybrid variants of GA have been proposed in the literature. Juang (2004) proposed a hybrid variant of GA by combining GA and PSO, named as HGAPSO. The HGAPSO applies PSO and GA simultaneously in order to obtain the most optimal solution. In a view to solve local stagnation problems, Dorn et al. (2011) combined GA with the structured population and hybridized with a path-relinking technique. GA has been successfully applied in facial emotion recognition applications. For example, Rizon et al. (2007) proposed a new fitness function for GA to determine the top lip and bottom lip features. Garg and Bajaj (2015) proposed a facial emotion recognition system which employed GA for reducing the features extracted by Independent Component Analysis.

Storn and Price (1997) proposed differential evolution (DE) algorithm, which is a vector based metaheuristic algorithm with good convergence property. DE also uses crossover and mutation techniques and can be considered as an improved version of GA with explicit updating equations. Another advantage of DE is that it uses real numbers as solution strings and does not need encoding and decoding. Aziza et al. (2014) applied DE algorithm for selecting the most significant facial landmarks points to improve the accuracy of facial emotion recognition. Bueno et al. (2013) proposed a Neural Evolution (NE) based facial emotion recognition algorithm. In the proposed NE, the DE algorithm was used to minimise the global error by tuning the weights and biases of the NN classifier.

The firefly optimisation algorithm was suggested by Yang (2008) to improve the global optimisation problem and find the optimal solutions by enabling fireflies with lower light intensities to move towards those with higher light intensities. The key concepts of the FA are as follows: (1) Fireflies must be Unisex; (2) fireflies with weak light intensity are attracted towards those with strong light intensities. (3) The light intensity of each firefly denotes the quality of the solution. The conventional PSO, DE and SA are considered as a simplified version of FA (Fister et al., 2013).

Over the years, many researchers have proposed improved variants of FA to enhance the performance and convergence speed of conventional FA. Discrete FA (DFA) was suggested by Sayadi et al. (2010) in a view to deal with NP-hard scheduling problems. Coelho et al. (2011) proposed a chaotic FA (CFA) which computes the inherent chaotic characteristics of FA under different parameter settings. Another interesting discrete variant of FA is Memetic FA (MFA), which was proposed by Fister et al. (2012) to solve combinatorial graph-coloring problems. A hybrid variant of FA was suggested by Abdullah et al. (2013) which combined FA with DE to estimate parameters of the nonlinear biological model. The improved Levy's flight FA (LFA) model was proposed by Yang (2010), which uses Levy flight as randomization parameter. LFA was able to find the global optima with higher success rate when compared with GA and PSO. This combination enabled the algorithm to avoid getting stuck in local optima. Abdullah et al. (2012) combined the conventional firefly algorithm with evolutionary operations of DE, which improved the searching accuracy and interaction between the fireflies. This algorithm divides the original population into two subpopulations, where FA was applied to the first sub-population and evolutionary operators of DE is applied in the second population. Verma et al. (2016) proposed opposition and dimensional based FA, where candidate solutions were initialized using opposition based learning to improve the convergence rate and the position of each firefly was updated along different dimensions to solve the high-dimensionality problem. Alweshah et al. (2015) hybridised LFA by employing simulated annealing (SA) to balance the exploration and exploitation. LSFA was applied to solve the classification problems and outperformed the conventional LFA and SFA models.

Another well-known population-based evolutionary algorithm, named ABC was proposed by Karaboga (2005). It simulates the natural foraging behaviour of a honey bee colony to solve the optimisation problems, where food sources represent solutions and nectar amounts represent fitness values. Over the years, many researchers have proposed different variants of ABC to improve the conventional ABC. For example, a binary ABC was proposed by Wang et al. (2010) to search optimum features for intrusion detection systems. Another variant named discrete ABC (DisABC) was suggested by Kashan et.al (2011), which measures the dissimilarity between binary vectors. Rajasekhar et.al (2011) applied a Levy probability distribution mutation technique to improve the exploration of the conventional ABC. The CS (Yang, 2009) is another nature inspired metaheuristic algorithm, which simulates the natural social behaviour of cuckoo birds. The CS algorithm employs Levy random step to explore the search space instead of simple isotropic random walks. The empirical findings show that CS algorithm outperforms GA, conventional PSO, and other state-of-the-art algorithms. Naik and Panda (2016) proposed an adaptive CS (ACS) without using Levy step. In conventional CS, the step size cannot be controlled, but ACS introduced a mechanism to control step size which is proportional to fitness of each nest.

Among these EC Algorithms, Particle Swarm Optimisation (PSO) algorithm is population-based metaheuristic algorithm proposed by Kennedy and Eberhart (1995). The PSO algorithm is inspired by the social behaviour of animals such as fish schooling and birds flocking. PSO has been widely applied in image processing domain (Niu & Shen, 2006; Omran et al., 2006) and shows better optimisation capability in comparison to GA (Silva et al., 2002; Esmin & Lambert-Torres, 2012; Zhao et al., 2005). Initially, PSO starts with a randomised population, which contains a pre-initialised number of particles. The main advantage of PSO is that it is easy to implement and does not require any gradient information. In PSO, the solution of the problem is formulated as a search space and each position in the search space is considered as the solution to the problem. Particles share information with each other to select the best position (best solution) in the search space.

There are many PSO variants proposed in the literature to overcome the local optimum problem of conventional PSO (Campos et al., 2014). Alviar et al. (2007) proposed

a best rotation PSO, which divides a single swarm into sub-swarms to optimise the multi-model functions. The best rotation PSO avoids the stagnation on local minima by forcing populations to move from one local minimum to another. This process increases the exploration of the problem space between multiple local minima. In adaptive PSO algorithm proposed by Xie et al. (2002a, 2002b), the inactive particles are adaptively replaced by new particles while preserving the relationships among the particles. Another variant of adaptive PSO was proposed by Zeng et al. (2006), whose search process is guided by Acceleration Information. This concept of acceleration item is employed while updating the position and velocity of each particle to improve the global search capabilities of PSO. Kennedy and Eberhart (1997) introduced a binary PSO in which changing the position of a particle in the search space equals to changing the probability of the fact that the value of position coordinate is 0 or 1. Pampara et al. (2005) proposed an angle modulated PSO algorithm. A trigonometry function is used to generate a bit string, and it shows better computational efficiency in comparison with binary PSO. Mahmoodabadi et al. (2014) proposed a PSO variant known as HEPSO. In HEPSO, PSO is integrated with a multi-crossover mechanism of the GA and the food source finding the operator of bee colony optimisation for updating the particle velocity and position, respectively. Evaluated with well-known benchmark functions, HEPSO has shown superiority over other PSO variants. Li et al. (2015) proposed another hybrid PSO algorithm with the integration of fuzzy reasoning and a weighted particle to guide the swarm. The weighted particle is used to adjust the search direction, whereas other parameters such as the attraction factor and inertia weight controlled by fuzzy reasoning are used to adjust local exploitation and global exploration to guide the search. The proposed model was tested with ten benchmark functions and was further applied to nonlinear neural network-based modelling. Jordehi (2015) proposed an enhanced leader PSO model known as ELPSO. ELPSO employs Gaussian, Cauchy, opposition-based, and DE based mutation to increase the diversity of the swarm leader.

PSO variants have also been extensively used for feature selection. Zhang et al. (2015) extended the conventional bare bones PSO (BPSO) to feature selection problems with binary variables. Known as binary BPSO, a reinforced memory strategy is used to

update *pbest* of each particle to retain swarm diversity, whereas a uniform combination technique is applied to increase local and global search capabilities of the algorithm. In binary BPSO, the influence of the uniform combination is strengthened as the occurrence of stagnated iterations of the algorithm increases. Wang et al. (2013) proposed a parameter-free Gaussian bare-bones DE algorithm (GBDE). GBDE employs Gaussian distribution as the mutation strategy and a self-adaptive scheme for crossover probability adjusting. GBDE has been further enhanced by integrating with DE/best/1 (another mutation strategy) to achieve a fast convergence rate. The enhanced model outperformed several DE variants and bare-bones algorithms.

Chuang et al. (2011) proposed chaotic binary PSO (CBPSO) for feature selection. It combines two chaotic maps, i.e. logistic and tent maps, with BPSO to determine the inertia weight, in order to overcome the local optima problem. The results indicate that CBPSO in combination with a tent map can produce the best performance. Xue et al. (2013) proposed two PSO-based multi-objective feature selection algorithms, i.e. NSPSO and CMDPSO, to generate a Pareto front of non-dominated solutions. NSPSO integrates the concept of non-dominated sorting with PSO, while CMDPSO embeds PSO with the strategies of crowding, mutation, and dominance. Both algorithms apply a crowding distance to the non-dominated solutions for maintaining the selected *gbest* diversity for each particle. Specifically, CMDPSO employs an external archive to store the non-dominated solutions and a binary tournament selection to generate *gbest* for each particle based on the crowding distance. It also uses the mutation operation to diversify the search. Evaluated with 12 datasets, CMDPSO outperforms NSPSO and other multi-objective algorithms, including non-dominated sorting genetic algorithm II (NSGAI).

2.3 Action Unit Intensity Estimation and Emotion Recognition

In recent years, Neural Networks (NN) is taking part in a significant role for variants of information mining tasks. The intention of the neural network algorithm is to mimic the human reasoning ability to acclimate to variable circumstances for problem-solving. Beginning from McCulloch-Pitts network, the analysis is extremely standard to a number of the upper order neural network. The methodology of a Neural Network is to imitate a

few capabilities of the human brain, which has incontestable perspective for numerous low-level computations and embodies outstanding features like learning, fault tolerance, similarity, etc. NN has been regarded as a standard technique and has been widely applied in most of the data mining fields as well as classification (Gish 1990; Michie et al. 1994; Zhang 2000), functional approximation (Belli et al. 1999; Castro, Mantas, and Benítez 2000; Funahashi 1989), foretelling (Adya and Collopy 1998; Callen et al. 1996; Church and Curram 1996; Faraway and Chatfield 2008; Fletcher and Goss 1993; Hippert, Pedreira, and Souza 2001; Weizhong Yan 2012), rule extraction (Andrews, Diederich, and Tickle 1995; Castro, Mantas, and Benitez 2002; Setiono, Wee Kheng Leow, and Zurada 2002), pattern recognition and medical applications (Hosseini-Nezhad et al. 1995; Lisboa 2002; Portney and Watkins 2015). Recently, ANN has stood forward as a robust variation to the standard recognition models. The computer science research is already in young age for the implementation technique of some standard ANN models like Multilayer Perceptron, Hopfield network, Self-Organizing Feature Map, Radial Basis Function, Learning Vector Quantization, Back Propagation Network Cellular Neural Network, Counter Propagation Networks, Adaptive Resonance Theory Networks, and Support Vector Machines etc. Because of the presence of these neural networks, it is expected a fast increase in our understanding of artificial neural networks resulting in improved network paradigms and a number of application opportunities.

The insidious use of Support Vector Machine (SVM) in numerous data processing applications makes it an essential tool within the development of merchandise that has implications for the human society. SVMs, being computationally powerful tools for supervised learning, are widely employed in classification, clustering and regression problems. SVMs are with success applied to a range of real-world issues (Burges 1998) such as particle identification, face recognition, text categorization, bioinformatics, civil engineering and electrical engineering, etc. SVMs as originally introduced by Vladimir Vapnik (Vapnik 1998) within the world of statistical learning theory and structural risk minimization, have demonstrated to work with success on numerous classification and prediction problems. SVMs are utilised in several pattern recognition and regression estimation issues and applied to the issues of dependency estimation, forecasting and

constructing intelligent machines (Ekici 2012). We have further discussed the applications of NN and SVM classifiers in facial AU and emotion recognition systems in the following.

Shan et al. (2009) focused on deriving facial feature representations using LBP and examined several LBP-based methods, including template matching, SVM, linear discriminant analysis, etc., for person-independent facial expression recognition. Their study indicated that in comparison to Gabor wavelets, LBP was able to extract features from a raw image with low computational cost, and was also able to store sufficient discriminative facial information in a compact representation. The work also further employed AdaBoost in order to retrieve the most discriminative LBP features. In comparison to LBP features, the Boosted LBP features were able to improve the recognition performances of different classifiers. It concluded that the best recognition performance was obtained by using SVM with Boosted-LBP features, although it showed limitations on generalisation to other datasets. Tsalakanidou and Malassiotis (2010) also proposed a real-time 2D+3D facial feature tracker based on ASM. The model used local appearance and surface geometry information to extract 81 facial points. Special trackers were also produced to further analyse the texture of the mouth and eyebrows. A rule-based approach was developed to detect four facial expressions and 11 AUs using the derived geometric and textural features. Zheng et al. (2008) also made attempts to improve the original ASM on several aspects including extending the profile of the original ASM from 1D to 2D in order to improve its fitting accuracy and efficiency in real-time applications. However, although the new model outperformed the original ASM for facial feature extraction, both of the ASMs were not able to deal with pose variations.

Although AU intensity estimation has been a challenging task and not very well explored, there is some recent development worthy of note mainly Kaltwang et al. (2012), Savran et al. (2012), and Li et al. (2013). Kaltwang et al. (2012) proposed a method of automatic continuous pain intensity estimation from facial images to benefit health care applications. A set of relevance vector regressions (RVRs) for continuous pain intensity estimation was trained with diverse geometric and texture feature sets. Evaluation using the recently published UNBC-MacMaster Shoulder Pain Expression Archive Database showed that the proposed feature fusion scheme outperformed those separately trained

RVRs with different feature sets. That is, the RVR with the combined texture features obtained via LBP and discrete cosine transform achieved the best performance. Li et al. (2013) proposed a dynamic Bayesian network to model dependence among spontaneous facial AUs and other related temporal behaviour for AU intensity measurement. Their work outperformed other image-driven AU estimation methods.

Facial emotion recognition from 3D inputs and video sequences has also been explored in the field. Li et al. (2010) presented a probabilistic neural network based 3D facial expression recognition system, which took not only localised facial feature points but also derived geometric features, such as slopes and angles, into account for facial emotion recognition. Majumder et al. (2014) developed a facial emotion recognition system based on geometric features using an extended Kohonen self-organizing map for the recognition of six basic emotions. The system automatically generated a geometric feature-based 26-dimension input vector with the consideration of landmarks for eyes, lips and eyebrows. Their proposed approach outperformed other popular classification schemes such as radial basis function (RBF) network, multi-layered perceptron and multi-class SVM. Fang et al. (2014) proposed a novel scheme for automatic dynamic salient information (such as peak expressions) extraction and facial expression analysis from video sequences. The proposed facial emotion recognition system outperformed static emotion recognition systems. Instead of depending on AU detection, their system studied a dynamically extracted feature space of over 300 dimensions for emotion recognition. Six state-of-the-art machine learning techniques were also used for system experiments and evaluation to prove the system's efficiency and robustness. They have also conducted user studies to investigate the correlation between human perception and their system's outputs.

Various attempts have been made to develop pose and illumination invariant emotion recognition classifiers. For example, Moore and Bowden (2011) proposed a multi-class SVM-based approach for facial emotion and pose classification. Their approach investigated variations of LBP-based descriptors and the influence of pose on diverse expression recognition tasks. LGBP features were especially evaluated in their work using images with multi-views. Chen et al. (2008) also proposed a hybrid boost learning mechanism to deal with face detection and emotion recognition against pose variations.

Skin colour detection and segmentation algorithms were used first to locate face region. Subsequently, both weak and vigorous hybrid classifiers were used for face detection and emotion classification against the scale and pose variations, and partial occlusions. Especially, they employed local Harr-like features with global Gabor attributes for the selection of the weak hybrid classifiers. Yu et al. (2013) also employed an SVM with active learning to collect facial images with realistic expressions from web sources. The database generated had more diversity with collected images much closer to real-world conditions in comparison to other well-known (e.g. CK+) databases. A novel feature descriptor was also employed in their research. Their work showed great potential to advance research on robust facial expression recognition. Moreover, as mentioned earlier, in cognitive studies, Martinez and Du (2012) proposed a new theoretical model for the description of multiple compound emotion categories such as happy surprise in order to overcome the difficulty that traditional emotion models encountered. This new theoretical model pointed out directions for building new computational models for compound emotional facial behaviour recognition. It also motivates us to employ FCM clustering to enable a test facial expression instance to belong to more than one emotion cluster and thus recognise compound emotions. The detailed discussion is presented in Chapter 4.

Krishna et al. (2014) developed a face recognition system with a method called Threshold-based Binary PSO Feature Selection (ThBPSO). ThBPSO conducts multi-runs of conventional BPSO and stores *gbest* identified from each run. Then, a threshold is used to determine the importance of each dimension of the global best solutions. A feature is selected and considered as important if the total number of selections of this feature in the past runs is more than the pre-defined threshold. The system was tested with seven benchmark datasets and showed superior performance over other state-of-the-art methods. Liu et al. (2015) proposed a deep learning architecture, i.e. Action Units inspired Deep Networks (AUDN), for learning facial expression features. AUDN consists of three sequential processes, i.e. (i) a convolutional layer and a max-pooling layer to learn the Micro-Action-Pattern (MAP) representation; (ii) feature grouping to integrate correlated MAPs to produce mid-level semantics; (iii) a multi-layer learning process to construct sub-networks for higher-level representations.

Zavaschi et al. (2013) proposed a novel facial expression recognition system with the integration of ensemble classifiers trained on both Gabor and LBP features. A set of 73 base SVM classifiers was generated by varying parameter settings of Gabor filters and LBP. NSGAI was used to identify the most optimal ensemble structures whose fitness function focused on the minimization of both error rate and the number of selected base classifiers in the ensemble. Diao et al. (2014) proposed an adaptive ensemble reduction technique by applying the heuristic harmony search (HS) algorithm. HS identified an optimal ensemble size while preserving or increasing ensemble diversity and classification accuracy.

Zeng et al. (2006) proposed a one-class classification system using Kernel whitening and Support Vector Data Description to distinguish spontaneous emotional expressions from outlier non-emotional expressions. Meng and Bianchi-Berthouze (2014) developed a multistage framework to explore continuous emotion recognition from naturalistic facial and vocal expressions where temporal relationships between consecutive levels of a given effective dimension were modelled using Hidden Markov Model (HMM). Regarding automatic multimodal emotion recognition, Zeng et al. (2007) conducted spontaneous emotion detection from audio-visual modalities using Adaboost multi-stream HMM. Soleymani et al. (2015) performed continuous emotion recognition from electroencephalogram (EEG) signals and facial expressions. The power spectral density from EEG signals and facial landmarks were employed to represent multimodal emotional inputs. Diverse regression models such as recurrent neural networks and continuous conditional random fields were used for emotion regression of the valence dimension.

Eleftheriadis et al. (2015) proposed a discriminative shared Gaussian process latent variable model (DS-GPLVM) for multi-view and view-invariant classification of facial expression. A discriminative manifold was derived based on learning of multiple views of a facial expression. Emotion classification was conducted using both the expression manifold and the view-invariant or multi-view information. Their work compared favourably with other related state-of-the-art developments. Happy and Routray (2015) proposed a facial expression recognition system with the consideration of texture features of selected salient facial patches. Active facial patches associated with emotional

expressions were initially extracted, which were then further analysed to obtain discriminative salient facial features for distinguishing between each pair of emotion classes. A facial landmark detection technique to enable more accurate localization of facial patches with less computational costs was also proposed. The system employed the one-against-one classification method for emotion recognition.

2.4 Summary

In this chapter, related work discussions are presented to cover facial feature extraction techniques, feature selection/optimisation algorithms and facial emotion recognition models. Motivated by the above literature review, we aim to propose a robust facial emotion recognition system to deal with challenges encountered during real-life applications. In Chapter 3, we employ a novel AAM combined with BRISK algorithm in order to improve the real-time face fitting and tracking of the AAM algorithm. Then, in Chapter 4 we employ Gabor filtering, a novel feature descriptor, BRISK, the ICP algorithm and FCM with post correlation processing for effective facial point detection from images with pose and scale variations, illumination changes, occlusion and background clutter to benefit real-life human-computer interaction. We also use regression-based AU intensity estimation and probability-based fuzzy clustering to measure the strength of 18 AUs respectively and recognise seven basic emotions, neutral expressions and compound emotions. Lastly, in Chapter 5 we propose a facial expression recognition system using a novel variant of LBP for discriminative feature extraction and micro-GA embedded PSO-based feature optimisation.

Chapter 3 Supervised Facial Feature Extraction Using AAM in Combination with BRISK

In this chapter, we present an intelligent facial action and emotion recognition system which employs supervised facial feature extraction model. Motivated by the FACS (Ekman et al., 2002), this research focuses on the recognition of seven basic emotions and 18 AUs. Since effective facial representations of the face images play crucial role in automatic facial expression recognition, this chapter employs a novel shape and appearance feature extraction method, which integrates an Independent AAM with a rotation-invariant feature point detector, BRISK (Leutenegger et al., 2011). In comparison to AAM with a traditional inverse compositional fitting, our model with BRISK fitting is computationally faster and is robust in dealing with feature extraction from images of faces with rotations and scaling differences without prior training required. Subsequently, shape and appearance based NN AU analysers are used to detect 18 AUs respectively. Emotions are then decoded from the derived AUs using an NN emotion recognizer. Evaluation results indicate its high accuracy for AU and emotion recognition. It is also among the top performers on the CK+ database in comparison to other existing state-of-the-art applications.

3.1 Feature Extraction

Feature extraction is the first step towards accurate and robust facial expression recognition. In this chapter, we have employed AAM as the base facial shape and texture feature extractor. This section provides a detailed discussion of original AAM, Independent AMM with inverse compositional fitting, and the proposed AAM algorithm combined with BRISK.

3.1.1 Introduction to active appearance model

In this section, we introduce the basic theory of AAM before we apply the AAM for shape and texture feature extraction. Cootes et al. (2001) proposed the original AAM algorithm. The AAM algorithm indicated to be very efficient and robust for deformable objects tracking in many applications such as medical imaging and facial geometric feature

extraction. According to Matthews and Baker (2004), there are two types of linear AAMs: independent and combined AAMs. In independent AAMs, the linear modelling of shape and appearance of deformable objects is performed separately. In comparison to independent AAMs, combined AAMs employ a single set of linear parameters to describe shape and appearance. The training of an AAM requires a set of images and coordinates of landmarks associated with all the training images as inputs.

For the building of independent AAMs, first of all, PCA (Swets & Weng, 1996) is used and applied to the training images to obtain the base shape (i.e. the mean shape) \vec{s}_0 and a linear combination of n shape vectors \vec{s}_i .

$$s = \vec{s}_0 + \sum_{i=1}^n (p_i \vec{s}_i) \quad (3.1)$$

Equation 3.1 (Matthews & Baker, 2004) is used to compute the model shape s , where p_i represents the shape parameters. Similarly, PCA is also applied to shape normalized training images to obtain the base appearance \vec{A}_0 and the appearance images \vec{A}_i . For example, if the model appearance image $A(x)$ is defined over the pixels within the interior of the base mesh \vec{s}_0 , the Equation 3.2 (Matthews & Baker, 2004) is used to compute the model appearance $A(x)$, where $\vec{A}_0(x)$ represents the base appearance with $\vec{A}_i(x)$ indicating the appearance images and λ_i representing the appearance parameters.

$$A(x) = \vec{A}_0(x) + \sum_{i=1}^m (\lambda_i \vec{A}_i(x)) \quad (3.2)$$

In AAMs, a Delaunay method is usually used to perform triangle texture warping of the base shape \vec{s}_0 and the model shape s respectively. In order to map the vertices of each triangle in s_0 with those of the corresponding triangle in s , a piecewise affine warp, $W(x; p)$, is defined. Thus the final AAM model instance is generated by forward warping the appearance A from \vec{s}_0 to s . For a combined AAM, a third PCA is subsequently applied to the training shape and appearance parameters, p and λ , obtained through an independent AAM via the above modelling procedures.

AAM fitting algorithm obtains the best fitting between the trained model and the given input image. Many strategies and algorithms for fitting have been proposed in the past to reduce the error between the given input image and the calculated model. The approach of updating the shape and appearance parameters respectively with Δp and $\Delta \lambda$ is the most intuitive method. However, research studies showed that it may not always lead to convergence and also require intensive computation (e.g. for the Lucas-Kanade algorithm) which makes it less efficient for real-time facial feature analysis.

Another comparatively more efficient and computationally affordable approach is to update the model with an incremental warp, $\vec{s}_0 W(x; \Delta p)$. There are two warp compositional algorithms: forwards and reverse compositional algorithms. The research carried by Matthews and Baker (2004) suggests that the inverse compositional algorithm is regarded as the more effective warp compositional algorithm for AAM fitting. Comparing to the forwards compositional algorithm, the inverse algorithm reverses the roles of the template and example input image and computes the incremental warp in the opposite reverse direction from the input test image to the template $\vec{A}_0(x)$.

3.1.2 Facial feature extraction using AAM

In this chapter, we have first applied AAM algorithm in order to capture discriminative facial features embedded in both facial motion and texture deformation. The feature extraction using AAM with inverse compositional fitting is the first step towards extracting discriminative facial features.

The images from the CK+ Facial Expression Database (Lucey et al., 2010; Kanade et al., 2000) are used to perform the training of the AAM. In order to build a robust and efficient AAM, we employ 2,926 images donated by 32 subjects extracted from the CK+ database for the training of AAM. (The subject IDs of the selected images from the database are from S005 to S065.) These training images represent frame-by-frame transitional facial behaviours from neutral faces to the peak emotional expressions. These training images and the corresponding 68 two-dimensional landmarks for each image are used as inputs to the training algorithm of AAM.

The building of AAM has been conducted in the following way. The landmarks associated with the training images provided by the database are used to build the shape point distribution model. As mentioned in Section 3.1.1, PCA is applied to the training images to obtain the mean shape \vec{s}_0 , a linear combination of shape vectors \vec{s}_i and shape parameters p_i . Then PCA is also applied to the shape normalized training images to generate the texture distribution model including the calculation of the mean appearance \vec{A}_0 , a linear combination of the appearance images \vec{A}_i and the appearance parameters λ_i . Then AAM instantiation takes place which employs the obtained shape and appearance parameters to generate the shape and appearance models, s and A . The final model instance is generated by warping the appearance A from \vec{s}_0 to s . As discussed earlier, the computationally intensive procedures for the effects of appearance variation during fitting in the inverse compositional algorithm can be pre-computed. Therefore, in the training algorithm of AAM, we also include the preparation steps for the fitting procedure for the inverse compositional algorithm including the calculation of the gradient of texture using the mean texture \vec{A}_0 , the warp Jacobian at the base shape and the modified steepest descent images in order to add appearance variations. 2,926 images extracted from the CK+ database are employed respectively for the training of AAM and the pre-computation of appearance variation for the inverse compositional algorithm. The trained AAM with the shape and texture information and piecewise affine warp values for model instantiation are then stored in a text file. The pseudo-code of the training algorithm of AAM is provided in Algorithm 3.1.

Algorithm 3.1. Training of an Independent AAM (Matthews & Baker, 2004)

Input: 2,926 training images and their corresponding landmark files with each file containing 68 landmarks for each image.

Output: A text file to store the trained AAM including:

1. A mean shape vector, eigen shape values and vectors
2. A mean texture vector, eigen texture values and vectors
3. Model Instantiation

begin

Step 1: Load the face detection classifier

Step 2: Train AAM and the inverse compositional algorithm

repeat

For each image and its landmark file

{

//The building of AAM

2.1 Build shape point distribution model, s

2.2 Build texture distribution model, A

2.3 Model Instantiation

//The pre-computation for inverse compositional algorithm

2.4 Calculate the gradient of texture using the mean texture

2.5 Calculate warp Jacobian at the base shape

2.6 Calculate the modified steepest descent image in order to add appearance variation

2.7 Calculate the inverse Hessian matrix using the modified steepest descent images

}

Until the last image and its landmark file

Step 3: Write the trained AAM to a text file

End

The trained AAM stored in the output text file includes a mean shape vector with 68 dimensions and a mean texture vector with 167,199 dimensions. The output file also contains the following information for each of the 2,926 training images: a shape eigen vector, a texture eigen vector, a shape parameter and a texture parameter. The related piecewise affine warp values for model instantiation are also stored in this output file.

For the testing of AAM, the facial emotion detection system provides real-time image alignment and fitting of AAM for any test input image sequence. The generated AAM using Algorithm 3.1 and images captured from the web-cameras are used as inputs to the fitting algorithm at the trial stage. First of all, the face detection algorithm (i.e. the Haar-based cascade classifier) will be used to detect facial regions. Then the system retrieves the mean texture and mean shape vectors generated for the trained AAM and the pre-computed inverse of the Hessian produced by Algorithm 3.1. The inverse compositional algorithm is subsequently used to perform image warping, image differencing, and image dot products. It also estimates the corresponding changes to the base mesh, Δp and the incremental warp, $W(x; \Delta p)$. The final outputs of the fitting stage are the generated best fitted shape

landmarks and an extracted texture model of the input test image. Algorithm 3.2 shows the fitting algorithm of AAM with the inverse compositional algorithm.

Algorithm 3.2. Fitting using the Inverse Compositional Algorithm with the Trained Independent AAM (Matthews & Baker, 2004)

Input: AAM generated using Algorithm 3.1 and images captured by the camera.

Output: The bested fitted shape landmarks and an extracted texture model of the input test image.

begin

Step 1: Load the AAM model generated using Algorithm 3.1.

Step 2: Load the face detection classifier

repeat

Step 3: Start the camera and capture images

Step 4: Search faces using the face detector

Step 5: Apply fitting on the detected face

```

{
    5.1 Load the mean texture from the trained AAM
    5.2 Check for the current shape
    5.3 Warp the image to the shape mesh
    5.4 Compute the error image and dot products
    5.5 Apply the inverse compositional algorithm to update the parameters
    {
        5.5.1 Estimate the corresponding changes to the base mesh,  $\Delta p$ 
        5.5.2 Compose the incremental warp with the current warp estimate,  $W(x; \Delta p)$ 
    }
    5.6 Update the warp and calculate the new constrained shape,
         $W(x; p) \leftarrow W(x; p) \circ W(x; \Delta p)^{-1}$ 
}

```

Step 6: Draw the output of the best fitted shape mesh vector with 68 dimensions on top of the input image.

Step 7: Display the output texture model on the terminal using OpenCV

until *Esc key is pressed or while loop ends*

end

Example outputs of the extracted shape and texture representations of an input test image using the inverse compositional algorithm with AAM are shown in Figure 3.1.

An inspection of the images from the CK+ database indicates that there are various minor head rotations from one frame to another. Also in order to deal with head rotations

and pose variations in wider real-world applications and perform more robust and efficient face tracking and geometric feature extraction, the POSIT algorithm is employed in this research to deal with head rotations. POSIT was initially proposed by DeMenthon and Davis (1995), which deals with finding the pose of an object from a single image. It uses an approximate pose to compute the scaled orthographic projections of feature points. The 68 output landmarks by AAM are thus further adjusted using POSIT by taking the pose of an object in an image into account to further improve fitting accuracy.

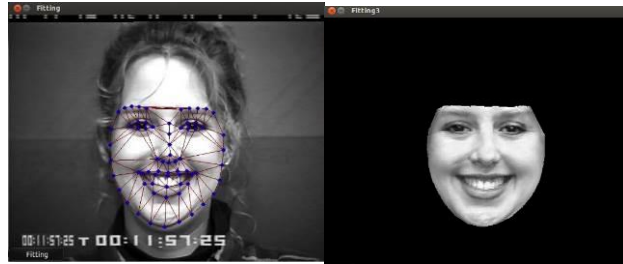


Figure 3.1. AAM outputs for a test image using the inverse compositional algorithm (the left image showing the generated facial mask with the derived 68 landmarks and the right one indicating the derived texture output of AAM)

Then the 68 adjusted output landmarks derived from the above POSIT based fitting are used as inputs to an NN classifier to detect facial AUs. These landmarks can better describe facial motions and thus capture more discriminative information for facial emotion recognition. Since the performances of AAM algorithm depends on fitting accuracy the IC and POSIT algorithms are not enough to achieve the best performance. The combination of IC and POSIT is computationally intensive and can only deal with head rotation up to 35° (Mistry et al., 2014) since it will become incredibly difficult to detect facial emotions if the head rotation is above 35° because of dramatic information loss.

3.1.3 Facial feature extraction using AAM and BRISK

So far we have applied AAM algorithm in order to extract the most discriminative facial feature from the input image. It also enables us to further improve its current fitting model to produce more efficient shape representations of original faces for automatic facial expression analysis.

Moreover, we employ an Independent AAM in this research, which is built from scratch. It performs linear modeling of shape and appearance of deformable objects and employs two sets of parameters respectively for shape and appearance feature extraction. Since the training of an AAM requires a set of images and coordinates of landmarks associated with these images, in this research, images from the CK+ facial image database (Kanade et al., 2000; Lucey et al., 2010) are used to build the AAM. The training images and their corresponding 68 2D landmarks are both used as inputs to the training of AAM.

The past research and our initial experiments show that the AAM with IC fitting is sensitive to rotation, pose variations and fast facial movement. Moreover, in order to effectively extract shape and appearance features from rotated images, it also requires intensive re-training of the AAM with database images of faces with rotations and pose variations. Such a training process is often time-consuming and less efficient and requires access to a large number of database images with diverse head rotations. Sometimes, this is even infeasible and computationally costly to achieve. In order to deal with these challenges, a more efficient fitting approach is required.

Several novel feature detectors and descriptors have drawn our attention because of their competitive computational speed and great flexibility in keypoint detection. SURF (Speeded-up Robust Features) (Bay et al., 2008), a scale and rotation-invariant interest point detector and descriptor, is first studied. It employs a Hessian matrix-based measure for the detector, and a distribution-based descriptor and also optimises these methods to the essential in order to achieve competitive performance and accuracy. BRIEF (Binary Robust Independent Elementary Features) (Calonder et al., 2010) is another feature point detector and descriptor. Although it easily outperforms other fast descriptors such as SURF regarding speed and recognition rate. However, BRIEF lacks rotational invariance.

BRISK (Leutenegger et al., 2011) is another rotation and scaling invariant method for keypoint detection, description and matching. Evaluation with benchmark datasets showed its computational efficiency and high-quality performance compared to other state-of-the-art detectors. It had impressive performances in dealing with rotations and scaling differences and proved especially suitable for tasks with limited computational power and time constraints. Therefore, it is selected for this research to perform real-time facial feature

extraction. Furthermore, BRISK has embedded a novel scale-space Fast Retina Keypoint (FAST)-based detector. It computes brightness comparisons from an easily configurable circular sampling pattern to generate a binary descriptor string. The combination of FAST with the assembly of this bit-string descriptor enables BRISK's fast performance. It overcomes the inherent difficulty and balances well between high-quality feature extraction and low computational cost (Leutenegger et al., 2011). Therefore, this research integrates BRISK with AAM to propose a rotation invariant feature extraction model as the novel contribution. The details are discussed below.

First, we apply the trained AAM in real-time and extract the initial shape information for a test image using the inverse compositional algorithm (IC). Then the IC algorithm terminates, and we focus on the building of BRISK. The shape information extracted from AAM is converted into two keypoint vectors k_1 , and k_2 where k_1 , represents the previous frame key-points and k_2 represents the current frame key-points. We then compute two BRISK descriptors using these generated key-point vectors. After that, we use an Approximate Nearest Neighbour (ANN) based descriptor matcher to find the match between these two generated BRISK descriptors. The match found after the ANN search is used as the final output for the shape information. Subsequently, texture features are also extracted by AAM with the guidance of these generated shape landmark outputs, i.e. the bested fitted 68 2D landmarks. The extracted texture features by AAM and BRISK have also been further represented by LGBP, which provides more discriminating power of the extracted texture features. These optimized geometric and texture features are used for the subsequent AU detection and emotion recognition in our experiments. Figure 3.2 shows the integration details of AAM with BRISK for appearance and shape feature extraction.

AAM with BRISK fitting also shows more computational efficiency (9-12 frames per second) in comparison to AAM with the inverse compositional algorithm (7-10 frames per second). By integrating BRISK with AAM in the fitting process, it enables our feature extraction model to deal efficiently with test images with rotations, pose variations and scaling differences without any prior training needed. It also significantly improves the accuracy of facial landmark point tracking, thus increases the accuracy for corresponding texture feature extraction. Trained with frontal facial images from CK+, AAM with BRISK

fitting has also been evaluated with rotated images from another (PUT) image database for facial feature extraction. The evaluation results in section 3.3 show that the overall system can efficiently deal with challenging fitting tasks with pose variations and rotations. Some example feature extraction outputs are provided in the evaluation section.

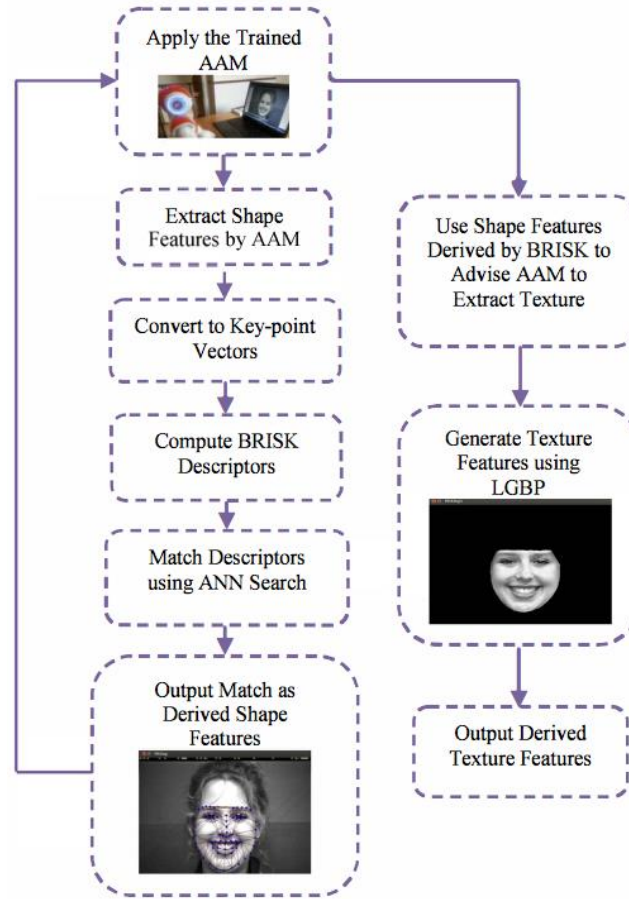


Figure 3.2. Shape and texture feature extraction using AAM with BRISK

3.2 Facial Action Unit and Emotion Detection

Optimal features will maximise between-class variations and in the meantime minimise inner-class variations. In this section, we test the efficiency of the derived shape and appearance features for AU and emotion recognition. Supervised neural NN classifiers with Back-propagation are employed to detect 18 facial AUs with the consideration of shape and appearance features respectively. Subsequently, emotions are further decoded from the derived best recommended AU combinations using an NN classifier.

3.2.1 Facial action unit intensity estimation

In this research, we focus on the detection of the following AUs, since based on the study of the facial image AU annotation provided by the CK+ database, these AUs are closely related to the expressions of the seven basic emotions: Inner Brow Raiser (AU1), Outer Brow Raiser (AU2), Brow Lowerer (AU4), Upper Lid Raiser (AU5), Cheek Raiser (AU6), Lid Tightener (AU7), Nose Wrinkler (AU9), Upper Lip Raiser (AU10), Lip Corner Puller (AU12), Dimpler (AU14), Lip Corner Depressor (AU15), Lower Lip Depressor (AU16), Chin Raiser (AU17), Lip Stretcher (AU20), Lip Tightener (AU23), Lip Presser (AU24), Lips Part (AU25), Jaw Drop and Mouth Stretch (AU26/27).

As mentioned above, both shape-based and texture-based NN's are used for AU detection. The final outputs of detected AUs are determined based on the intensity ranking of all the AUs generated by these two processes. NN's have been widely used for facial action and emotion recognition and other pattern recognition tasks (Zhang et al., 2013). Back-propagation is also one of the most popular learning techniques for multi-layer NN's. Also, an NN-based classifier has the following advantages: (a) using synaptic weights to store knowledge, (b) operating as nonlinear dynamical systems and (c) implicitly detecting complex relationships between dependent and independent variables. Thus it is selected for AU detection in this research.

In the first experiment, the NN classifier employs geometric features, i.e. the 68 landmarks generated by AAM and BRISK based fitting, as inputs and outputs the detected 18 AUs. Moreover, the derived 68 2D facial landmarks by the fitting process contain 51 points for a description of facial elements and 17 points for facial shape information. In detail, the 51 shape features include 5 points for each eyebrow, 20 points for the mouth contour (i.e. 12 for the upper lip and 8 for the lower lip), 6 points for each eye and 9 points to characterise the nose position. Each facial point is represented by a pair of x and y coordinates. The BRISK feature detector also provides efficient adaptive fitting to deal with pose variations, illumination changes and scaling differences.

In order to balance between the accuracy and computational overhead for NN parameter selection, experiments using three different parameter settings for NN were conducted:

- a) One input layer (136 input nodes), one hidden layer (18 nodes), and one output layer (18 nodes).
- b) One input layer (136 input nodes), two hidden layer (each with 18 nodes), and One output layer (18 nodes).
- c) One input layer (136 input nodes), three hidden layer (each with 18 nodes), and 1 output layer (18 nodes).

Compared to the setting ‘a’, the other parameter settings were computationally intensive but achieved a similar accuracy. Therefore, this research employs the shape-based NN AU analyser with one input layer with 136 input nodes representing the 68 2D landmark points, one hidden layer with 18 nodes and one output layer with 18 nodes indicating the intensities of the recognised 18 AU’s. The shape-based AU recognition is trained with 327 images from the CK+ database. These images represent the frontal view of peak frame emotional facial expressions with AU and emotion annotations provided.

In order to extract texture features with more discriminating power, LGBP is employed to represent the texture features generated by AAM and BRISK based fitting. LGBP integrates multi-scale and multi-resolution Gabor filters and LBP operator together. It first decomposes a facial image into Gabor Magnitude Pictures using Gabor filters and then applies LBP (Zhang et al., 2005). The features generated by LGBP are more robust and contains more competitive discriminating characteristics than those derived by other operators e.g. LBP or Gabor filters (Zhang et al., 2005; Yan et al., 2011). The extracted texture features of emotional facial behaviours are then applied for AU recognition.

A texture-based NN AU analyser is thus implemented to generate intensities of the recognised 18 AUs. The texture-based classifier also shares a similar network topology as the shape-based AU analyser with three layers. The texture vector produced by LGBP is used as inputs to the AU analyser, while the hidden and output layers of the analyser respectively contain 18 nodes for the representation of the 18 detected AUs. In this

research, the 327 frontal view images for the training of the shape-based AU detection are also used for the training of the appearance-based NN classifier.

The shape and texture based AU analysers are also evaluated with the first 200 images taken from the CK+ database. Since in CK+ the maximum number of AU annotations provided for one image file is seven, we select the top seven derived AUs with comparatively higher intensities respectively for the shape and texture-based AU analysers for system evaluation.

Since discriminative physical cues of emotional facial behaviours are best captured by both shape and appearance features, in order to further improve AU recognition accuracy, we combine the results gained from the texture and shape-based approaches and derives another set of AUs with comparatively higher intensities from the combined outputs in the following way. We first of all rank the derived AUs obtained from both of the approaches based on their strengths. Top seven AUs with comparatively higher intensities (i.e. stronger physical cues) are then selected as the joint outputs. Evaluation results indicate the AU detection results obtained from the combined outputs gain the highest accuracy in comparison to those generated by either shape-based or appearance-based analysis. The AU detection evaluation for the shape and appearance-based and the combined approaches is discussed in detail in Section 3.3.

3.2.2 Facial expression recognition

Emotions have also been further decoded from the 18 derived AUs. A supervised NN-based facial emotion recognizer is implemented to recognise the seven basic emotions from facial expressions represented by the 18 AUs. This emotion recognition classifier also has a three-layer topology with one input, one hidden and one output layer. It accepts the overall 18 upper and lower AUs as inputs and outputs the recognised seven basic emotions embedded in facial expressions. Therefore, it has 18 nodes in the input layer and seven nodes respectively in the hidden and the output layers. The seven detected emotions include ‘happiness’, ‘anger’, ‘disgust’, ‘fear’, ‘sadness’, ‘contempt’ and ‘surprise’. The emotion classifier is trained with 173 images with AU and emotion annotations from the CK+ database. We evaluate the NN-based emotion recognizer with input AUs derived

respectively by the shape-based, the appearance-based and the combined approaches for the first 200 images extracted from CK+ database in the evaluation section of this chapter.

3.3 Evaluation

The AAM and BRISK based feature extraction model, the shape and texture based NN AU detection and the NN-based emotion recognizer are all implemented from scratch in C++ programming language under the Ubuntu operating system. OpenCV library is used to enable real-time vision perception, i.e. the system produced by this research has been integrated with OpenCV library to perform real-time facial emotion recognition for both real testing subjects and database images.

For the evaluation of the system, testing usually starts with natural dialogue based interaction between the computer and the developer or testing subjects. This includes greeting, the introduction of the testing purposes, conducting real-time face tracking, facial emotion recognition and emotional gesture generation as a response. Either a frontal-view posed facial expression of a testing subject or a still image was taken from the CK+ database is shown in front of the computer. The recognised upper and lower facial actions and the embedded emotions in the facial expressions are eventually communicated back to the testing subjects by the speech synthesis engine.

In this research, the accuracy and robustness of the facial feature extraction model and AU and emotion analysers are evaluated using images from the CK+ database since the database provides sufficient AU and emotion annotations for performance comparison. Another reason to select the CK+ database is simply that many other state-of-the-art applications used this database for evaluation and this enables us to compare our results with other applications efficiently. Moreover, in order to demonstrate the efficiency of real-time facial feature extraction using AAM and BRISK, example images from the CK+ and PUT (Kasinski et al., 2008) databases are employed. The CK+ database only has 593 AU annotated images, and 327 emotion annotated images, which lead to a smaller training and testing sets.

The PUT database is especially designed to provide a benchmark dataset with a significant size and diversity to evaluate the efficiency and robustness of face pose

estimation and pose-invariant face recognition algorithms. Although the AAM is built with frontal face images from the CK+ database, the evaluation shows that the BRISK feature detector also enables the model to conduct adaptive fitting effectively for test images with diverse head rotations and occlusions (e.g. classes) from the PUT database. Figure 3.4 shows some example outputs for shape-based feature extraction for images taken from both CK+ and PUT.



Figure 3.4. Real-time feature extraction and shape fitting using AAM with BRISK (the first image is taken from CK+ database, and the other two are from PUT database)

As mentioned earlier, first 200 images in CK+ are used to evaluate the shape and texture-based AU detection. AAM and BRISK, first of all perform real-time fitting and extract shape and appearance representations of test images. Then the derived geometric and texture features are respectively used for AU detection. The derived top seven AUs with comparatively higher intensities are used for evaluation respectively for the shape, texture and combined approaches. Overall, we have achieved averaged accuracy rates of 75.22%, 80.78%, and 85.5% respectively for the shape, appearance-based and the combined approaches for AU detection. The detailed evaluation results for each AU are also provided in Table 3.1.

Table 3.1. Action Unit Detection results for shape based, texture based and combined approaches

| AUs | Accuracy of the shape-based approach (%) | Accuracy of the appearance-based approach (%) | Accuracy of the combined approach (%) |
|---------|--|---|---------------------------------------|
| AU1 | 83 | 82.22 | 83 |
| AU2 | 85 | 85.9 | 85 |
| AU4 | 75 | 79.56 | 79 |
| AU5 | 73 | 94.35 | 95 |
| AU6 | 66 | 89.35 | 91 |
| AU7 | 76 | 77.32 | 79 |
| AU9 | 88 | 78.82 | 89 |
| AU10 | 55 | 64.11 | 67 |
| AU12 | 89 | 80.71 | 91 |
| AU14 | 65 | 75 | 85 |
| AU15 | 75 | 74.36 | 75 |
| AU16 | 57 | 76.34 | 78 |
| AU17 | 91 | 78 | 90 |
| AU20 | 82 | 52.76 | 84 |
| AU23 | 52 | 86.61 | 87 |
| AU24 | 64 | 89.81 | 90 |
| AU25 | 94 | 92.2 | 94 |
| AU26/27 | 84 | 96.59 | 97 |
| Average | 75.22 | 80.78 | 85.5 |

As shown in Table 3.1, among the 18 facial actions, Inner and Outer Brow Raiser (AU1 and AU2), Nose Wrinkler (AU9), Lip Corner Puller (AU12), Chin Raiser (AU17), Lip Stretcher (AU20), Lips Part (AU25), Jaw Drop and Mouth Stretch (26/27) have especially been well captured by the shape-based AU analyzer, since the above facial actions are more closely related to either shape or position changes of facial elements (e.g. Inner and Outer Brow Raiser, Lip Corner Puller, and Lips Part) during emotional expressions. Especially in comparison to the appearance-based approach, Nose Wrinkler (AU9), Lip Corner Puller (AU12), Chin Raiser (AU17), and Lip Stretcher (AU20) are better detected by the shape-based NN AU analyser.

Moreover, evaluation results indicate the appearance-based approach shows better performance for the following AU detection: Inner and Outer Brow Raiser (AU1 and AU2), Upper Lid Raiser (AU5), Cheek Raiser (AU6), Lip Corner Puller (AUI2), Lip Tightener (AU23), Lip Presser (AU24), Lips Part (AU25), Jaw Drop and Mouth Stretch (26/27). In comparison to the shape-based approach, the appearance-based detection shows great improvements for the detection of the following AUs: Upper Lid Raiser (AU5), Cheek Raiser (AU6), Upper Lip Raiser (AUI0), Dimpler (AUI4), Lower Lip Depressor (AUI6), Lip Tightener (AU23), and Lip Pressor (AU24). These AUs proved to be difficult to be identified purely by the shape-based detection since they tend to have less dramatic or very mild facial shape changes (e.g. Lower Lip Depressor (AUI6) and Lip Tightener (AU23)). However, most of them are more likely to indicate physical cues such as appearance deformations (e.g. Dimpler (AUI4), Lip Tightener (AU23)). Thus the texture based approach is more reliable for the detection of such AUs. Overall, results indicate that the combined AU recognition achieves the highest detection accuracy.

The NN-based facial emotion recognizer has also been tested against the affect annotation provided by the CK+ database. The AU sets respectively derived by the shape, texture and the combined approaches are employed to decode emotions embedded in test images. As mentioned earlier, the NN-based emotion classifier is trained with 173 images from the CK+ database. Evaluations results for the testing of first 200 images extracted from the CK+ database indicate that the system achieves the highest accuracy of 91.65% for the detection of the seven basic emotions using the AUs derived by the combined approach, while it respectively obtains 71.71% and 90.7% accuracies using AUs respectively derived shape and appearance based detection. We have conducted the experiments using AAM with the IC algorithm for AU and emotion recognition for the performance comparison with this research as well. The same set of 200 test images from CK+ is also used for the evaluation of the AAM with the IC fitting. Overall, the AAM with the IC fitting has achieved averaged accuracy rates of 67.37%, 78.7%, and 81.73%

respectively for the shape, texture and the combined approaches for AU detection. AAM with IC also achieves an averaged accuracy rate of 88.83% for the detection of the seven emotions using the AUs derived by the combined approach and accuracies of 70.1% and 83.39% using AUs respectively derived by the shape and texture based methods.

Table 3.2. Facial expression recognition results comparison between this research and related work

| | AAM+IC (%) | Shan et al. (2009) (%) | Wu et al. (2010) (%) | Zhang & Tjondronegoro (2011) (%) | This work (%) |
|------------------|------------|------------------------|----------------------|----------------------------------|---------------|
| Anger | 83 | 85 | 83 | 87 | 87 |
| Disgrace. | 90 | 98 | 68 | 90 | 94 |
| Fear | 93 | 80 | 67 | 92 | 90 |
| Happy | 87 | 98 | 88 | 98 | 97 |
| Sadness | 96 | 75 | 78 | 91 | 93 |
| Surprise | 93 | 97 | 88 | 100 | 97 |
| Contempt | 75 | - | - | - | 85 |
| Neutral | - | 92 | - | - | - |
| Average | 89(7) | 89(7) | 79(6) | 93(6) | 92(7) |

The above AU and emotion detection results indicate our research using AAM and BRISK based feature extraction outperforms the system with traditional AAM fitting based on the IC algorithm. Moreover, this research shows performance improvements for the shape-based, appearance based and the combined approaches for both AU and emotion recognition with less computational costs in comparison to the system built based on AAM and the IC algorithm.

We also compare the best emotion detection results of this research gained from the combined approach with the most optimal results gained using AAM with BRISK and other state-of-the-art applications in Table 3.2. These state-of-the-art developments are selected for comparison because of the employment of the same testing strategy and testing database. Shan et al. (2009) employed Boosted Local Binary Pattern (LBP) based SVM to recognise six basic emotions and neutral with the evaluation conducted using images from the CK+ database. Wu et al. (2010) employed spatiotemporal Gabor filters for automatic

facial emotion recognition. Images from CK+ were also used to evaluate their system's performance as well. Zhang and Tjondronegoro (2011) conducted patch-based 3D Gabor feature extraction to perform facial emotion detection.

Evaluation results shown in Table 3.2 indicate the shape and texture combined emotion detection outperforms both the related research using Gabor filters and Boosted-LBP features (Shan et al., 2009; Wu et al., 2010) and AAM with the IC fitting. Furthermore, our research is also comparable to the work of (Zhang & Tjondronegoro, 2011). Based on the evaluation of the six emotions without the consideration of contempt, our work achieves the same accuracy, 93%, as that of (Zhang & Tjondronegoro, 2011). The proposed shape-based feature extraction model can achieve a similar accuracy as those obtained by the texture based models (e.g. Zhang & Tjondronegoro, 2011) while balancing between performance and computational efficiency. The results also indicate the effectiveness of the proposed AAM and BRISK-based feature extraction and the shape and texture combined facial AU and emotion recognition.

3.4 Summary

In this chapter, we have implemented supervised feature extraction based facial action analysers and an NN-based facial emotion recognizer respectively to detect 18 AUs and seven basic emotions from real-time posed facial expressions and database images. A novel feature extraction model with the integration of AAM and BRISK is proposed to conduct efficient face fitting and rotation invariant feature extraction with less computational cost. It shows competitive performances comparison to other state-of-the-art research. Since facial expressions for each emotion category have their unique characteristics, in future work, we aim to employ optimisation algorithms to extract further optimised discriminating texture features attached to each emotion category to better deal with real-time facial emotion recognition. Facial AU knowledge will also be employed to deal with landmark generation for images with occlusions.

Chapter 4 Unsupervised Facial Point Detection and Emotion Recognition

In this chapter, we discuss the proposed unsupervised feature point detection for intelligent facial emotion recognition. We propose unsupervised automatic facial point detection integrated with regression-based intensity estimation for facial AUs and emotion clustering to deal with scaling differences, pose variations and occlusions. The proposed facial point detector can detect 54 facial points in images of faces with occlusions, pose variations and scaling differences using Gabor filtering, BRISK, the ICP algorithm and FCM. Especially, in order to effectively deal with images with occlusions, ICP is first applied to generate neutral landmarks for the occluded facial elements. Then FCM is used further to reason the shape of the occluded facial region by taking the prior knowledge of the non-occluded facial elements into account. Post landmark correlation processing is subsequently applied to derive the best suitable geometry for the occluded facial element to further adjust the neutral landmarks generated by ICP and reconstruct the occluded facial region. We then conduct AU intensity estimation respectively using support vector regression (SVR) and NNs for 18 selected AUs. FCM is also subsequently employed to recognise seven basic emotions as well as neutral expressions. It also shows great potential to deal with compound and newly arrived novel emotion class detection. The overall system has also been evaluated with challenging real-life facial emotion recognition tasks to prove its efficiency.

4.1 Overview of the Proposed Methodology

The system includes key steps including face and ROI detection, unsupervised feature point detection, AU intensity estimation and emotion clustering. First of all, an improved version of Viola and Jones (2001) algorithm for face detection is applied to the input image. The three cascade classifiers are employed to detect the region of interest (ROIs) successfully. The detected ROIs consist of both eyes along with eyebrows, the nose tip and mouth. To further improve the automatic facial point detection, the following pre-processing methods and techniques are employed. The bilateral filter is employed to reduce

noise present in the input image. Then to extract the contour of each ROIs a 2D Gabor filter is applied to each ROIs. After applying the bilateral and Gabor filters, we can extract 16 facial points including four landmarks points for each of the ROI. However, these 16 facial points do not include any landmark points generated for the eyebrow. We employ robust, fast performance feature descriptor BRISK in order to extract number of facial points for each ROI. The BRISK model provides key improvements to the system, such as scale and rotation invariant descriptor and detector, and ability to deal with image key-point detection without sufficient prior knowledge on the scene and camera poses. The outputs generated by BRISK is further integrated with the edge detection results from the Gabor filtering to generate 21 facial points.

In order to further improve and capture more detailed facial points the ICP algorithm is employed where 21 facial points obtained from BRISK and Gabor filtering are regarded as reference points and a set of 68 landmarks of any neutral image borrowed from the CK+ Facial Expression Database (Kanade et al., 2000, Lucey et al., 2010) is used as a source point cloud. It reconstructs the 2D mesh and outputs the 54 detected facial points. However, the ICP model can only reconstruct the neutral occluded facial element, which makes it less efficient for occluded facial images. The FCM clustering algorithm is applied with the prior knowledge of the attributes of the non-occluded facial regions as inputs to further reason the shape of the occluded facial element. The 54 facial points derived from above are then used to estimate intensities of the 18 selected AUs. To estimate the intensity for each AU, we employ 18 SVRs and 18 NNs respectively. Then the FCM clustering is employed for emotion recognition to recognise basic and compound emotions. The detailed data flow of the system is also provided in Algorithm 4.1. The overall system architecture with facial point detection for non-occluded images is provided in Figure 4.1 whereas the system architecture with facial point detection for images with occlusions is presented in Figure 4.2.

Algorithm 4.1. Intelligent Facial Point Detection and Emotion Recognition

Input: (1) A test facial image

(2) A landmark files with 68 landmarks for any neutral image provided by the CK+ database

Output: (1) AU intensity of the selected 18 AUs

begin

repeat

Step 1. Load the test input image and the source input landmark file with 68 landmarks taken from the database.

Step 2. Process the test input image for feature point extraction.

For each image or frame

{

2.1 Conduct face and ROI detection.

2.2 Increase the contrast of each ROI.

2.3 Apply the bilateral filter to reduce the noise of each ROI.

2.4 Apply the Gabor filter on each ROI to detect its contour.

2.5 Apply the BRISK keypoint detector to extract landmarks.

2.6 Use the generated 21 facial points obtained from the combined outputs from 2.4 and 2.5 as the reference point cloud and transform the source 68 neutral landmarks from CK+ to best match the above 21 reference points using the ICP algorithm to produce a set of detected 54 facial points.

2.7 If (images contain occlusions)

{

2.7.1 FCM is applied to further inference the shape of the occluded facial element.

2.7.2 Post landmark correlation processing is then applied to derive the best fitting geometry for the occluded facial element to adjust the neutral landmarks generated by ICP and output the final 54 detected landmarks.

}

2.8 Output and plot the generated 54 landmarks on the test image.

}

Step 3. Measure AU intensity based on the detected 54 facial landmarks using SVRs and NNs.

Step 4. Use FCM to conduct emotion clustering.

Until *ESC key is pressed*;

End

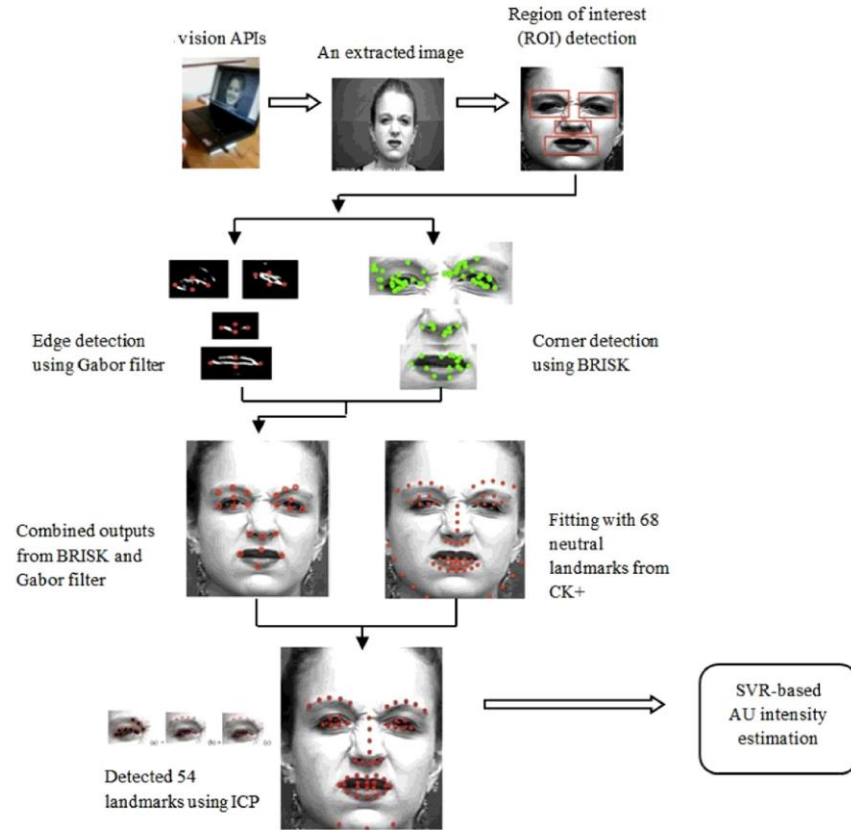


Figure 4.1. Overall system architecture with facial point detection for non-occluded images.

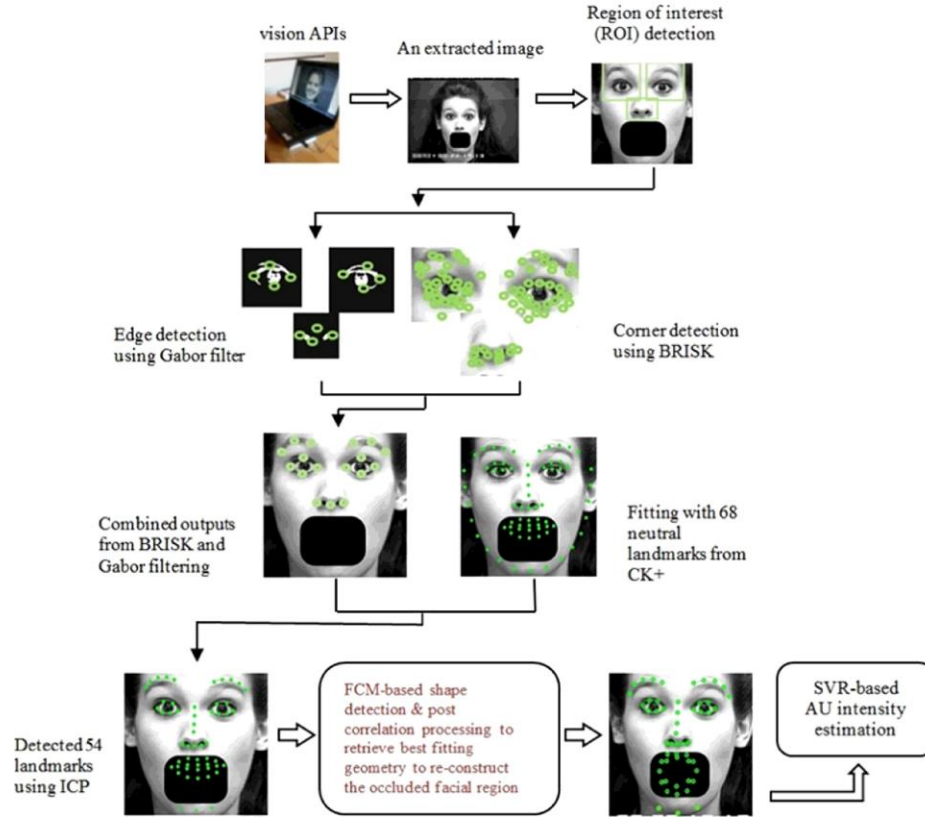


Figure 4.2. Overall system architecture with facial point detection for occluded images.

4.2 Unsupervised Facial Point Detection

As discussed in Chapter 2, the supervised facial feature detection models such as AAM, ASM and CLM rely on a large number of training data. Due to training with specific databases, it makes them less efficient when evaluated with different databases (Jaiswal et al., 2013). For example, if the supervised AAM model is trained with only frontal-view face images then it will not be able to extract the feature efficiently from non-frontal or multi-view images from other databases. Moreover, these models tend to require the higher computational cost to perform real-time face fitting (Jaiswal et al., 2013; Cristinacce & Cootes, 2006). Thus, to address the above challenges, we have implemented an unsupervised robust facial point detector. The proposed unsupervised facial point detector focuses on dealing with the challenging tasks such as landmark detection in pose variations, illumination changes, occlusion and background clutter when compared with the above-

mentioned supervised models. The proposed facial point detector model consists of various advanced feature extraction algorithms and extracts high-quality features while keeping low computational costs. The step by step representation of the proposed model is illustrated in Algorithm 4.2.

Algorithm 4.2. Unsupervised facial point detection

Input:

- (1) A test facial image
- (2) A source landmark file with 68 landmarks for any neutral image from the CK+ database

Output: 54 facial points for the test image

Begin

Repeat

- 1. Load the test input image from video inputs and the input landmark file with 68 landmarks taken from CK+.
- 2. If (the image and landmark files not loaded) exit (0).
- 3. Load Haar Cascade classifiers for face and regions of interest (i.e. areas of eyes, nose and mouth) detection.
- 4. Conduct feature point extraction.

For each image or frame

{

- 4.1 Convert the input image into a gray scale image.
- 4.2 Equalise the histogram values of the converted gray scale image to increase contrast.
- 4.3 Apply Haar face detector and extract the face region.
- 4.4 Apply Haar Cascade feature detectors on the detected face region to extract ROIs.
- 4.5 Increase the contrast of each ROI.
- 4.6 Apply the bilateral filter to reduce the noise of each ROI.
- 4.7 Apply the Gabor filter on each ROI to detect its edge and derive the initial 16 facial points.
- 4.8 Apply the BRISK feature detector and extend the detected number of landmarks to 21.
- 4.9 Apply the ICP algorithm to obtain a set of 54 detected landmarks.

{

- 4.9.1 Retrieve the previously loaded source 68 neutral landmark cloud provided by the database.

4.9.2 For each ROI, apply the ICP algorithm to reserve 2D facial curves and reconstruct 2D surface with the corresponding source neutral landmark cloud and the corresponding detected reference facial point cloud obtained from 4.8 as inputs.

}

- 4.10 If (Occlusion occurred)

{

```

4.10.1 Apply FCM to inference the shape of the occluded facial region.
4.10.2 Apply post landmark correlation processing to derive the best fitting geometry for the occluded
facial element and adjust the neutral landmarks generated by ICP.
}
4.11 Output the generated 54 landmarks of the test image.
}
Until ESC key is pressed;
End

```

4.2.1 Face and region of interest detection

In this research, we have employed an improved version of the Viola and Jones' algorithm (Viola and Jones, 2001) to detect the face accurately and locate the region of interests in the input image. We conduct pre-processing of the input image in order to improve the accuracy of face detection algorithm. First, we convert the input image into grayscale and a histogram equalisation method is applied in order to improve the contrast ratio of the gray-scale image. We then implement the improved Viola and Jones' algorithm to detect the face region accurately from the pre-processed image. The original version of Viola and Jones (2001) face detection algorithm was based on Adaboost algorithm. In order to improve computational efficiency in real-time application, Lienhart and Maydt (2002) replaced Adaboost with Gentleboost. This improved version of the face detection algorithm is employed in all our research for facial expression recognition.

In order to extract features from each important region of the face, ROI detection is performed. The three region-specific cascade classifiers (right and left eye cascade classifier and lower face cascade classifier) borrowed from OpenCV library (Castrilln et al., 2007; Castrilln et al., 2008) are employed in order to detect all the ROIs. As stated in (Castrilln et al., 2007; Castrilln et al., 2008), these three classifiers were respectively trained with a large amount of positive and negative images in order to detect each ROI efficiently. The process of detecting ROI is presented as follows:

In order to gain more visibility of important region of the face (eyes, nose and mouth), we apply histogram equalisation method to the detected face region. The detected face

region is further divided into three sections including areas of the right and left upper faces and a lower face region in order to retrieve ROIs with high accuracy (see Figure 4.3).

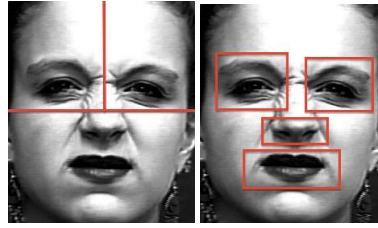


Figure 4.3. Face division (left) and ROI detection (right)

Then the three cascade classifiers for ROI detection are respectively applied on these three facial parts. The ROIs recovered include the positions of both eyes along with eyebrows and locations of the nose and the mouth (see the right diagram in Figure 4.3).

The three cascade classifier for ROI detection achieved 100% accuracy when tested with 1000 images collected from CK+ database. This ROI detection successfully detects the ROIs from real-time videos with up to 60-degree rotations. In order to detect the occluded region, we employ the following strategy. For example, for a test facial image where the mouth area is occluded, first we detect the face using the Viola and Jones face detector and after detecting the face, we apply Haar-cascades to detect each ROI. As discussed above, we employ three cascade classifiers respectively for ROI detection of both eyes along with eyebrows and a lower facial region. For a test image with the mouth area occluded, while detecting the ROI, the occluded mouth area is not detected by the Haar-cascade detectors. In other words, the region of the face is occluded if ROI detection does not detect that specific region. Section 4.2.4 presents a detailed discussion on the reconstruction of the occluded facial region. Also, the Gabor filter and BRISK algorithms will only be applied to the non-occluded facial region.

4.2.2 Gabor filtering based feature extraction

The detected ROIs are then further processed by using bilateral and Gabor filtering in order to extract the corresponding borders and contours. First, each ROI is processed using a bilateral filter to reduce the noise. In comparison to other filtering algorithms (such as a normalised box filter and a median filter), bilateral filtering is not only a nonlinear and noise-reducing smoothing process for images, but also edge-preserving (Tomasi &

Manduchi, 1998; Paris et al., 2008). The weighted average of intensity values from nearby pixels is used to replace the intensity value of each pixel in order to reduce the noise while preserving the sharp edges. Bilateral filtering has been applied separately on each ROI. The main reason behind applying a bilateral filter is to reduce noise in the input image and increase the accuracy of the Gabor filtering based edge detection for each ROI.

Later, we employ Gabor filtering in this research to extract the contour of each ROI. A Gabor filter is a linear filter and is widely used for edge detection and rotation sensitive local frequency detectors in computer vision field. The Gabor filters are popular for texture segmentation analysis due to its optimal localization properties in both spatial and frequency domains. In this research, we employ a 2D Gabor filter to extract the edge from each ROI. The research presented in (Donato et al., 1999) shows that the Gabor filtering based feature extraction outperforms PCA, local feature analysis and Fisher's linear discriminant. Most importantly, Gabor filters can remove the light and contrast variations in images while preserving their robustness. The following equations define the 2D Gabor filter employed in this research. These include simple definitions for the real and imaginary Gabor kernel generation without DC compensation.

Real:

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = e^{\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)} \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (4.1)$$

Imaginary:

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = e^{\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)} \sin\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (4.2)$$

$$x' = x \cos \theta + y \sin \theta \quad (4.3)$$

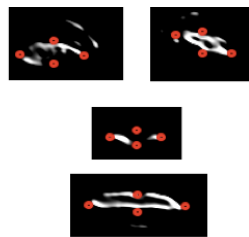
$$y' = -x \sin \theta + y \cos \theta \quad (4.4)$$

Where λ is the wave length of the sinusoidal factor with θ as the anti-clock wise rotation of the Gaussian and the plane wave, ψ as the phase offset, σ as the sigma/standard deviation of the Gaussian envelope, and γ as the spatial aspect ratio (Daugman, 1988).

In this research, we apply the Gabor filter on each ROI to detect its edge. Figure 4.4 shows the sample output images generated by Gabor filter. The detected outlines include the important regions such as eyes, mouth and nose. These outputs are Gabor filter magnitude image, where the detected important edge regions are in white colour with the rest of black color pixels. Also, in order to recover four keypoints for each ROI based on these filtering outputs, we connect the biggest and closest blocks of white pixels to form a big block and draw a rectangle around it. We then subsequently assign a key-point respectively on the left and right side of the rectangle and the midpoint of the upper and lower side of the rectangle (see Figure 4.4). Thus we recover overall 16 key-points for eyes, nose and mouth with each of these facial elements allocated four facial points. This 2D Gabor filter is also able to deal with head rotations effectively. However, it is not able to recover any keypoints for both eyebrows, which may play important roles in emotional facial expressions (e.g. AU1 and AU2 are present for surprised emotion).

Figure 4.4. Edge detection using a 2D Gabor filter

In order to increase the system's robustness, identify facial features points for both eyebrows and further improve the accuracy of above detected 16 facial points, we employ a robust novel keypoint descriptor and detector, BRISK. This keypoint detector is used to extend the current detected facial landmarks further and is also robust enough to deal with scaling differences, pose variations and head rotations. We discuss this keypoint detector in the next section.



4.2.3 Facial point detection using BRISK

In computer vision, several feature point detector and/or descriptor are introduced in order to deal with automatic feature extraction. We have explored some of the well-known techniques and packages in order to find the most suitable facial feature detector for this research. The detectors and descriptors we explored include SURF (Bay et al., 2008),

BRIEF (Calonder et al., 2010), FREAK (Ortiz, 2012) and BRISK (Leutenegger et al., 2011). Introductions of these packages are provided below.

First of all, SURF is a scale and rotation-invariant interest point detector and descriptor. It employs a Hessian matrix-based measure for the detector, and a distribution-based descriptor and also optimises these methods to the essential to achieve competitive computational speed and accuracy. Upright SURF (U-SURF) is a scale invariant only version of this descriptor (Bay et al., 2008).

The BRIEF feature point detector and descriptor rely on a relatively small number of intensity difference test to represent an image patch as a binary string. the Hamming distance is employed in order to compare strings. BRIEF prove to show better performance regarding speed and landmark detection when compared with SURF and U-SURF. However, BRIEF fails to deal with rotational invariance (Calonder et al.,2010).

FREAK is a retina inspired fast, compact and robust keypoint descriptor (Ortiz, 2012). Although, as a descriptor, it is more robust and fast to compute in comparison to SURF and BRISK (Ortiz, 2012), it is not a feature point detector as required by our research.

As discussed in Chapter 3, BRISK is another novel method for keypoint detection, description and matching. Evaluation with benchmark datasets proved its computational efficiency and high-quality performance compared to other state-of-the-art descriptors and detectors. A novel scale space FAST (Features from Accelerated Segment Test) based detector has been embedded in BRISK. Moreover, BRISK generates a binary descriptor string by computing brightness comparisons from an easily configurable circular sampling pattern. The fast performance of BRISK is obtained by combining FAST with the assembly of this bit-string descriptor from intensity comparisons. It overcomes the inherent difficulty and balances well between high-quality feature extraction and low computational requirement. Thus BRISK is suitable for real-time applications with limited computational power and time constraints (Leutenegger et al.,2011). Therefore, we have selected BRISK to perform real-time facial point detection to deal with challenging real-life human-machine interaction.

In detail, BRISK focuses on scale-space key-point detection. It overcomes the scale invariance problem by estimating the actual scale of each key-point in a continuous scale-space. A saliency measurement is also used to detect points of interest across both the image and scale dimensions. In order to achieve high computational efficiency, it detects points of interest in both octave layers of the image pyramid and layers in-between. The location and scale of each key-point is identified by using the quadratic function fitting. Especially it can deal with image key-point detection tasks without sufficient prior knowledge on the scene and camera poses. In (Leutenegger et al.,2011) it shows that BRISK is efficient and robust in dealing with rotation and scaling variations.

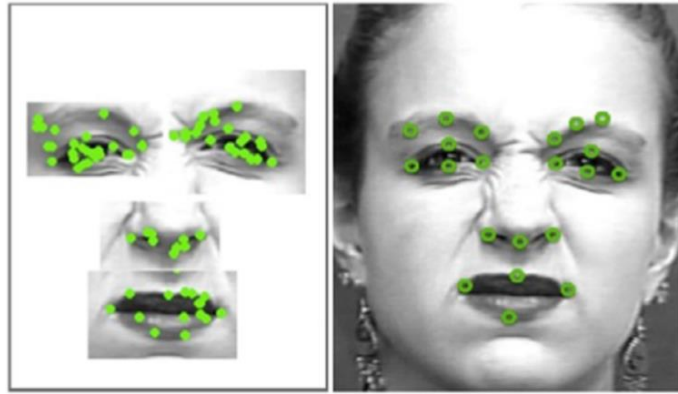


Figure 4.5. Keypoint detection for each ROI using BRISK (left) and the final detected 21 point (right).

We integrate a C++ version of BRISK for facial point detection in this research. Similar to Gabor filtering, BRISK is also applied directly to each ROI image to retrieve key facial feature points. The advantage of using BRISK is that it can derive numerous facial points with high accuracy. Since it is combined with a high-speed corner detector, FAST, BRISK can provide more reliable corner detection results in comparison to the 2D Gabor filter. It also shows high levels of repeatability under diverse changes of head poses, rotations and scales. The BRISK generates a sequence of landmark points ranging from 100-200 for all ROIs. An example output of BRISK is also provided in Figure 4.5. Since BRISK is used in our facial point detector, we can reliably deal with rotations and pose variations at least up to 60 degrees in real-time applications.

The Gabor filtering generates one core corner detection point for each potential candidate corner without any overlapping and redundancy. These detected points are used

as reference to the BRISK based feature detection. Thus, we combine these two sets of key-point outputs for each ROI produced respectively by the Gabor filter and BRISK in order to generate the final set of landmarks for each facial element and further increase the accuracy of feature point detection.

Algorithm 4.3. Output Combination of Gabor and BRISK

Input: (1) The outputs produced by Gabor filter and BRISK

Output: (1) 21 selected keypoints

begin

Step 1. //Reduce the number of feature points detected by BRISK

For each keypoint in BRISK's output

{

1.1 Apply circle-circle intersection method.

1.2 If (Keypoint is intersected by more than 5 times)

{

1.2.1 Consider it as an important keypoint.

} else

1.2.2 Remove the keypoint from the feature set.

1.3 Add the newly selected keypoint to the new BRISK feature output

}

Step 2. For each keypoint in new BRISK feature output and Gabor filter output

{

2.1 Apply circle-circle intersection method.

2.2 If (New BRISK feature output keypoint overlapping the Gabor filter output keypoint)

{

2.2.1 Use BRISK keypoint as the feature point.

} else

2.2.2 Remove the keypoint from feature set.

}

Step 3. Combine the eyebrow keypoints generated by BRISK with the above new output.

End

The step-by-step process of combining the landmarks generated by BRISK and Gabor filter is given in Algorithm 4.3. Firstly, the landmarks detected using BRISK and Gabor filter are represented by using small circles. The circle-circle intersection algorithm

(Weissstein & Eric, 2015) is employed in Algorithm 4.3 step 1 in order to reduce the number of feature points detected by BRISK. Thus a set of important features is collected from the output of BRISK by selecting the feature points with the highest number of overlappings. Again in step 2, the circle-circle intersection technique is also applied to find out overlapping points in the newly generated BRISK output and Gabor filter output. Finally, the most accurate 15 features are generated after the combination of the two outputs. However, the step 1 and step 2 does not provide landmarks for eyebrows since features for eyebrows are not present in the output of Gabor filter. Therefore, in step 3, an extra set of six points with a highest number of overlapping's for the eyebrows generated by BRISK is integrated with the above recovered 15 landmarks in order to produce a set of 21 facial points as outputs. Algorithm 4.3 shows the details of the output combination of BRISK and Gabor filter using the circle-circle intersection method.

Especially, BRISK can retrieve landmarks for both eyebrows in comparison to the outputs generated by Gabor filtering. Eventually, 21 landmarks are generated based on the combination of the BRISK and Gabor filtering based facial point detection. These 21 generated landmarks include three points for each eyebrow, four for each eye, and three for the nose and four for the mouth contour.

However, even with advantages like scale and rotation invariance, this facial point detection also has limitations such as when partial facial images are not visible because of occlusions or structural disturbances (e.g. make-up, glasses, facial hair, mugs etc.), the above facial point detection processing may not be able to recover all of the essential facial points for each facial element. The ICP algorithm (Besl & McKay, 1992) and FCM is employed in order to deal with the occlusion problem.

4.2.4 Facial point detection for occlusion using ICP and FCM

The ICP algorithm is widely used for reconstructing 2D or 3D surfaces and restores 2D curves in computer vision research. It is an algorithm employed to minimise the difference between two sets of points. It uses one reference and one source point cloud as inputs. The source point cloud will be transformed to provide the best match to the reference set while the reference point cloud will remain unchanged (Besl & McKay,

1992). In order to generate source and reference point clouds, we use 21 combined landmarks generated by BRISK and Gabor filtering as the reference clouds and randomly selected set of 68 neutral landmarks from the neutral image provided by the CK+ database as the source point cloud. Since we only focus on 54 facial landmarks for a test image, we only select 54 landmarks from the 68 neutral landmarks by discarding 14 landmarks for the description of the overall facial contour. We then align the neutral landmarks on each ROI of the test image. The reason that the alignment for each ROI is done separately is to make the neutral landmarks best fitted with the test image.

Algorithm 4.4. Iterative Closest Point Algorithm (Besl & McKay, 1992)

Input:

(1) 21 reference landmark points generated by Algorithm 4.3 and neutral 68 source landmark points.

Output:

(1) 54 generated landmarks for the test image

begin

While (the maximum number of iterations is not met)

{

Step 1. To select each point in the source point cloud and each point in the reference point cloud for each ROI and match these points to compute the closest points.

Let S be the source point set with N_s points denoted as $\vec{s}_l: S = \{\vec{s}_l\}$ for $l = 1, 2, \dots, N_s$ whereas R be the reference point set with N_r points denoted as $\vec{r}_k: R = \{\vec{r}_k\}$ for $k = 1, 2, \dots, N_r$. Calculate the Euclidean distance metric, d , between each pair of points in S and R which is denoted as follows:

$$d(\vec{s}_l, R) = \min_{r \in R} \|\vec{s}_l - \vec{r}\| \quad (4.5)$$

By referring C as the closest point operator, the resulting set of closest points, Y , is obtained as follows:

$$Y = C(S, R) \quad (4.6)$$

Step 2. To estimate the transformation using a mean squared error in order to best align each source point to its match found.

After obtaining the closest point set Y , the least squares quaternion operation, Q as mentioned in (Besl & McKay, 1992), is used to compute the least squares registration as follows:

$$(\vec{q}, d_{ms}) = Q(S, Y) \quad (4.7)$$

where d_{ms} is the mean square point matching error and \vec{q} is the registration vector.

Step 3. To transform the source points using the obtained transformation (e.g. rotation and translation).

Update the positions of the data set S until it reaches the termination point. The update of positions is given as follows:

$$S = \vec{q}(S) \quad (4.8)$$

Where $\vec{q}(S)$ denotes the updated point set S after transformation by the registration vector, \vec{q} .

}

End

We employ ICP algorithm for each ROI in order to align the source neutral landmarks selected from the database to the reference points spawned by BRISK and Gabor filtering using iterative transformation to progressively reduce the distance between the source and the reference points. As mentioned earlier, we are going to preserve the positions of reference point cloud i.e. points generated by BRISK and Gabor filtering, while the positions of source point cloud, i.e. the neutral points provided by CK+, will be progressively updated to best match the reference point cloud. Firstly, ICP algorithm retrieves the closest point in the reference point set for each source point then it calculates rotation and translations between each pair of source and reference points in order to provide best alignment between them. Afterwards, it transforms the source points depending on the above estimation. It continues this process until it reaches the maximum number of iterations (Besl & McKay, 1992). The pseudo-code of the ICP algorithm with equations in reference to (Besl & McKay, 1992) is provided in Algorithm 4.4.

In this research, the ICP algorithm is employed in order to transform 54 source neutral landmarks to best align the 21 reference facial points. This process allows us to reconstruct 2D surfaces, while extending the number of detected facial points from 21 to 54. The resulting 54 facial points consists of 5 points for each eyebrow, 6 landmarks for each eye, 9 for the nose, 20 for the mouth, and 3 points for the chin. Figure 4.6 shows an example output of ICP algorithm. The overall detected 54 facial landmarks for example image retrieved after applying ICP algorithm is shown Figure 4.7.



(a) (b) (c)

Figure 4.6. (a) shows the combined output of BRISK and Garbor filtering. (b) shows the fitted 68 neutral landmarks provided by CK+. (c) shows the final set of 54 facial point outputs generated by ICP with (a) and (b) as inputs.

Although the ICP algorithm is able to reconstruct the missing facial features with promising performance in real-time applications, the facial landmarks generated by ICP always represent a neutral facial element. Therefore, we apply unsupervised learning technique FCM clustering in order to restore emotional expression for the occluded facial regions. In Section 4.3.2 we have explained the FCM clustering in detail. We introduce shape estimation using FCM for the occluded facial elements in the following.

Firstly, the whole face is divided into three different parts such as left upper face, right upper face, and lower facial region. Subsequently, three FCM algorithms are developed in order to recover the contours or shapes of the occluded left eye, right eye and the mouth. The landmarks of non-occluded facial regions are used as input for each FCM and it outputs three clusters representing an opened, narrowed and neutral facial element. For the retrieving the shape of the mouth, 22 landmarks representing both eye regions with 10 landmarks denoting eyebrows and 12 landmarks denoting eyes are used as inputs whereas for the inference of the shape of either eye, 20 landmarks which form the geometry of mouth and 11 points representing the other visible eye region are used as inputs. The three output clusters of FCM represent either mouth wide open or lip corner puller/closed neutral mouth or widened/tightened/neutral contour of the eye, depending on occluded facial region. The output shape information of each FCM is further used to adjust the neutral facial landmarks of the occluded element produced by ICP. The FCM and related post processing on top of ICP is used only if the occlusion is occurred in the test image otherwise the landmark generation completes after the application of ICP.



Figure 4.7. An example image from CK+ with the detected 54 facial points.

As discussed earlier, the test image is grouped into a specific shape clusters once the FCM has successfully predicted the shape of the occluded facial region. Then within this shape cluster, landmark correlations between the visible facial elements of the test instance and the corresponding facial elements of other samples in the cluster have been calculated. Then five images with the highest correlations to the visible facial elements in the test image are selected. The landmark points from these highly correlated five images for the corresponding occluded facial element in the test image (e.g. the mouth) are retrieved and then averaged to produce the landmarks for the occluded facial element in the test image.

For example, let us consider a facial image with surprise emotion has a mouth wide open and eye widened but the mouth is occluded and the intensities of how wide the mouth is open are unknown. We have used the above averaging method to normalise the shape obtained from FCM. In other words, the top five output shape cluster obtained after the application of FCM with the highest correlations to the test image in the same cluster are selected and averaged in order to re-construct the best fitting geometry for the occluded facial element. This reconstructed set of landmarks with shape information embedded is then used to replace neutral landmarks generated by ICP in the test image.

Section 4.4 presents detailed evaluation results of facial landmark generation for images with occlusions and rotations. Landmark detection using a combination of ICP and FCM with post correlation processing outperforms the landmark detection using only ICP for an image with occlusions. Related discussions are provided in Section 4.4.1. An example occluded image before and after applying FCM with post correlation processing

for the retrieval of landmarks for the occluded facial element (the first two images) with landmark generation for the same image without occlusion as a reference (the third image) is shown in Figure 4.8. It indicates that FCM with post correlation processing has further adjusted the set of neutral landmarks reconstructed by ICP to retrieve an opened mouth based on the inference of the non-occluded facial components and the retrieval of best-fitted geometry. Overall, ICP and FCM with post correlation processing enable us to reconstruct missing landmark points for a test image to effectively deal with occlusion.

The step-by-step procedures are also summarised in the following for the occlusion detection (steps 1 and 2) and landmark generation for the overall image (steps 3–6).

1. We apply cascade classifiers for ROI detection in order to identify the positions of the left eye, right eye, nose tip and mouth.
2. If any of the ROI is occluded, then the corresponding Haarcascade detector will not detect it. That is, the image occlusion is detected.
3. Only the detected ROIs are further processed using Gabor filter and BRISK to recover key landmarks for these facial regions. In another word, Gabor filter and BRISK are not applied to the occluded facial regions.
4. After applying Gabor filter and BRISK, we get a set of landmark points (less than 21) for the non-occluded facial elements (e.g. a total of 17 landmarks for a test image with mouth occlusion).
5. Then we employ a set of neutral landmarks with ICP to construct 54 landmarks for the overall image with neutral landmarks recovered for the occluded region (i.e. the mouth).
6. To further re-construct the landmarks for the occluded facial region, we employ FCM and some post correlation processing to identify the best fitting geometry, further adjust the shape of the occluded facial element and output the final set of 54 landmarks.

These detected landmarks for the overall image are subsequently used for SVR-based facial AU intensity estimation and FCM-based emotion recognition.

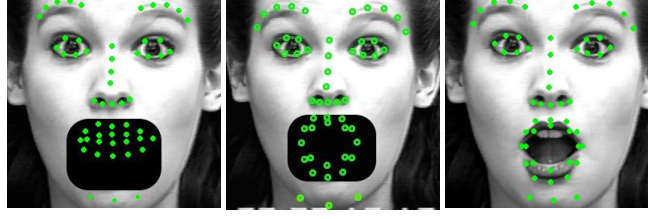


Figure 4.8. An example occluded image with landmarks generated by ICP (left) and further adjusted by FCM (middle) with landmark generation without occlusion as reference (right)

Moreover, we have evaluated this unsupervised facial feature point detection model using five image databases, i.e. CK+, LFW, PUT, LFPW and Helen, in order to prove its flexibility and computational efficiency. Detailed evaluation results for this facial point detection processing using the above five databases are discussed in Section 4.4.

4.3 AU Intensity Estimation and Emotion Clustering

The geometric features are widely used for facial action and emotion recognition research because of their effective capability of capturing physical cues effectively embedded in emotional facial expressions (Zhang et al., 2013; Li et al., 2010). In this research, we also employ the derived 54 facial points in order to estimate AU intensity and recognize facial expressions from test images. SVR and NNs are employed for estimating the AU intensity for selected 18 AU's. FCM clustering is also used to detect the eight basic emotions including happiness, anger, sadness, disgust, surprise, fear, contempt and neutral. Experiments of compound emotion recognition using FCM are also conducted.

4.3.1 AU intensity estimation

In this research, we only focus on 18 AUs out of 32 AUs defined by the FACS. These selected 18 AUs are closely related to the expression of eight basic emotions including Inner Brow Raiser (AU1), Outer Brow Raiser (AU2), Brow Lowerer (AU4), Upper Lid Raiser (AU5), Cheek Raiser (AU6), Lid Tightener (AU7), Nose Wrinkler (AU9), Upper Lip Raiser (AU10), Lip Corner Puller (AU12), Dimpler (AU14), Lip Corner Depressor (AU15), Lower Lip Depressor (AU16), Chin Raiser (AU17), Lip Stretcher (AU20), Lip Tightener (AU23), Lip Pressor (AU24), Lips Part (AU25), Jaw Drop and Mouth Stretch

(AU26/27). In order to measure the intensities of these 18 AUs, we have employed a dedicated SVR and NN for each AU making a total of 18 SVRs and NNs. The SVRs and NNs promising performances and robustness of the modelling of the problem domain makes them a very strong candidate for AU intensity estimation in this research. Detailed discussions and justifications are provided in the following.

SVR is a widely known nonlinear regression technique, which uses a nonlinear mapping to convert the original training data into a higher dimension and computes a linear regression function in this converted high dimensional feature space. SVR aims to identify the most suitable hyperplane, which can accurately predict the distribution of data within an error tolerance value of ε . SVR estimates the function of data instead of classifying data into two or more distinctive classes. In comparison to SVM it has crucial differences such as: (1) The predicted label value of an instance is a continuous value in SVR but a discrete one in SVC. (2) For SVR, there is a tolerable error, ε , between the predicted value and the actual label value. However, in SVM the predicted and actual class types must be either exactly matched or no matching at all. SVR has been applied in various fields to solve regression problems such as financial time series forecasting (Montana & Parrella, 2008).

In this work, an SVR derived from Libsvm (Chang & Lin, 2011) package integrated with OpenCV has been employed for AU intensity estimation. Libsvm is an efficient software package for SVM classification and regression. Epsilon-SVR from Libsvm is employed in this research. We construct 18 epsilon-SVRs in order to effectively estimate intensities for the 18 selected AUs. Facial elements related to each AU intensity estimation are also identified. We use 31 landmarks related to the upper face elements such as eyes, eyebrows and nose as inputs to the SVRs in order to measure the intensities of AU1, 2, 4, 5, 6, 7 and 9, while 20 facial points related to the mouth contour are used to estimate the intensities of lower facial AUs including AU10, 12, 14, 15, 16, 17, 20, 23, 24, 25, and 26/27.

We execute the linear scaling of the input values of each training example after the associated facial landmarks are determined for each AU intensity estimation. The Libsvm library offers the efficient scaling preprocessing, which scales each attribute value to the

range of [0, 1]. The same scaling method is also used to adjust the test data. This scaling process ensures that attributes in greater numeric ranges will not dominate those in smaller numeric ranges.

In this work, we explore both linear and non-linear RBF kernels in order to select the optimal kernel. In this research, we select the non-linear RBF kernel for SVR-based AU intensity. This is mainly because: (1) RBF nonlinearly maps inputs into a higher dimensional space, thus it can deal with the case that the relation between facial features and AU intensity levels is non-linear. (2) RBF has a fewer number of hyper parameters than other non-linear kernels (e.g. polynomial kernel), which may reduce the complexity of model selection (Hsu et al., 2010). (3) RBF usually has lower computational complexity, which leads to optimal real-time computational performance. Also, since the number of attributes for each AU intensity estimation is not very large, this makes the RBF kernel more suitable to this application domain in comparison to a linear kernel (Hsu et al., 2010). Thus an RBF kernel is selected for each of the 18 SVRs in order to achieve promising performances.

The most important part in setting up each RBF-based SVR is to identify the optimal kernel parameters. The three important parameters for an RBF-based SVR, whose values need to be determined including a soft-margin constant, C , a kernel parameter, gamma and an epsilon in the loss function. The combination of these three parameters plays very important roles in affecting each SVR's performance. In order to identify the most optimal set of the three parameters, a grid search on C , gamma and epsilon, and cross validation have been performed (Hsu et al., 2010). A tenfold cross validation has also been conducted in order to find the best combinations of parameters while avoiding over-fitting. We use exponentially growing sequences and employ values respectively ranging from 2^{-5} to 2^{15} for C , 2^{-10} to 2^5 for gamma and 2^{-8} to 2^{-1} for epsilon for the grid search. The search experiment is guided by the mean squared error (MSE) of each AU intensity estimation. The MSE value is computed using the following equation (DeGroot, 1980):

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - y_i^*)^2 \quad (4.9)$$

where y_i is the predicted value, and y_i^* is the original annotation.

The best parameter set of C , gamma and epsilon within the search space in our application is identified for each SVR, which yielded the lowest MSE for each AU intensity estimation. Moreover, the training and testing of the corresponding nonlinear SVR model is based on these identified most optimal parameters for each AU intensity estimation. As stated earlier, 18 SVRs are employed for AU intensity measurement of the 18 selected AUs. In this research, images from the CK+ database are used to perform the training and the grid search of these 18 SVRs. The extended CK+ database contains 593 sequences of frontal-view posed facial images contributed by 123 subjects. Each image also contains 68 2D landmark points stored in an individual text file. The database also includes 593 FACS coded peak frame emotional images and 327 peak facial images have also been annotated with the selected seven emotion labels including happiness, anger, sadness, disgust, surprise, fear and contempt (Kanade et al., 2000, Lucey et al., 2010). Therefore, we employ 200 FACS coded peak emotional images with AU intensity annotation and 50 neutral images from the CK+ database for the cross validation processing of these 18 SVRs. Moreover, a different number of images is also used for the training of each SVR based on the availability of the corresponding AU among the extracted 250 training images. It ranges from 15 images used for the training of intensity estimation for AU10 to 125 images used for intensity estimation for AU25. On average, 75 images are used to train each SVR.

Moreover, we use CK+ database with AU intensity annotation for evaluating the 18 SVRs because the previously selected databases for the evaluation of the facial point detection, i.e. PUT, LFW, LFPW and Helen, do not provide any AU and emotion annotation also with the majority of the images indicating neutral expressions. In order to evaluate the 18 SVRs, the 127 peak emotional images from the CK+ database with a fair distribution of each AU and 30 neutral images are used.

In order to evaluate the performance of SVRs, 18 feed-forward NNs with back-propagation to perform intensity regression for the 18 AUs are employed. These 18 NNs employ the same corresponding set of either 31 upper or 20 lower facial landmarks as inputs for the AU intensity measurement for the 18 AUs with each NN dedicated to the intensity estimation for each AU. Moreover, each NN model uses one single hidden layer because a single hidden layer can approximate any continuous functions. Also, each NN is trained with the same training set employed by the corresponding SVR for each AU intensity estimation. The same test set of 157 (127 emotional + 30 neutral) images extracted from CK+ is also used to test the performance of NNs.

The evaluation results in Section 4.4 indicated that SVR-based AU intensity estimation outperforms the NN-based measurement. SVRs show better performance for intensity estimation for the following AUs: AU1, 2, 4, 5, 6, 10, 12, 14, 15, 17, 20, 23, 25, and 26/27 in comparison to the NNs based estimation. The performance comparison between SVRs and NNs is discussed in detail in Section 4.4. Subsequently, these estimated intensities for the 18 selected AUs are used to infer emotions embedded in the real-time facial expressions using FCM.

4.3.2 Facial emotion clustering

In cognitive research, the perception of facial emotions was regarded to be based on a categorical model (Ekman & Rosenberg, 2005), which has been intensively employed in the machine learning field. In comparison with the categorical model, neuroscience research suggested that the perception of facial emotions was best to be described as a continuous model (Russell, 2003), where each emotion was described using characteristics common to all emotions in a multidimensional space. Although this model showed advantages in explaining emotion intensities compared to the categorical model, it was still not easy to use it to describe compound emotions. Therefore, Martinez and Du (2012) proposed a new theoretical model for the description of multiple compound emotion categories such as happy or angry surprise. Their model aimed to overcome the difficulty that both of the categorical and continuous models encountered. Their proposed method was to define N distinct continuous spaces and linearly combine these several spaces to

recognise compound emotion categories for facial expressions. This new theoretical model pointed out directions for building new computational models for the recognition of compound facial behaviour and also motivated this research.

Therefore, we employ an unsupervised FCM clustering technique to recognise the seven basic emotions and neutral expressions. It also shows great potential in detecting compound and newly arrived novel emotions to address the above challenge. Clustering algorithms usually organise objects into groups based on similarity criteria. The clustering techniques are categorised in three different methods each focusing on different clustering mining tasks. These three categories of clustering techniques can be stated as partitioning, hierarchical, density-based and grid-based clustering methods. However, the main drawback of these basic clustering algorithms is that they tend to force clustering an object into only one cluster. Sometimes, this rigid clustering may not perform as expected for some application domain such as, a compound facial emotion may belong to more than one emotion cluster and an online shopping review may contain comments related to several products, which may fall into several product clusters. Therefore, in order to overcome such problem, we employ probabilistic model-based clustering to allow one facial expression image represented by facial actions to be grouped into more than one emotion cluster (Han et al., 2011). It would be even more useful if a weighting is also calculated to reflect the strength of an object belonging to one cluster. Thus, a well-known probabilistic model-based clustering algorithm, FCM clustering, is used in this research to detect emotions.

Given a set of objects, o_1, o_2, \dots, o_n , and k pre-defined fuzzy clusters, C_1, C_2, \dots, C_k , FCM clustering groups each object into more than one cluster and generates a partition matrix, $M = [w_{ij}] (1 \leq i \leq n, 1 \leq j \leq k)$, where w_{ij} is the degree of membership that a data point belongs to a cluster. The partition matrix also needs to fulfil the following criteria (Han et al., 2011). (1) The degree of membership w_{ij} for each object o_i belonging to a cluster C_j , should be in the range of $[0, 1]$ to ensure that a fuzzy cluster is a fuzzy set. (2)

For each object, o_i , $\sum_{j=1}^k (w_{ij}) = 1$. This indicates that each object participates in the

clustering equivalently. (3) For each cluster, C_j , $0 < \sum_{i=1}^n (w_{ij}) < n$. This requirement is used to ensure that for each cluster, there is at least one object in this cluster with a non-zero membership value.

The expectation-maximization (EM) algorithm is employed to calculate a probability distribution over the clusters to obtain degrees of membership for each data object. It includes two procedures: the expectation and maximisation steps. The algorithm begins with initializing random parameters and iterates until the clustering cannot be improved (i.e. the clustering converges, or the change to the membership values between the most recent two iterations is sufficiently small). In each iteration, it calculates the centre of each cluster and updates memberships for fuzzy clustering. Both of expectation (E-step) and maximisation (M-step) steps are involved in each iteration. The E-step first of all selects random data instances as the initial centres of clusters. It then calculates degrees of membership for each data in each cluster. We consider the idea that if a data point, o_i , is closer to a cluster, C_j , the degree of membership of o_i for this cluster C_j should be higher. For one data object, the sum of its membership values for all the clusters will be 1. Thus, the E-step produces fuzzy memberships and partitions objects into each cluster. Equation 4.10 is used for the calculation of the degree of membership for a given data object x_i belonging to a cluster C_j (Han et al., 2011).

$$w_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (4.10)$$

where, m is the pre-defined fuzziness coefficient and C_j represents the center of each cluster. The fuzziness coefficient m , where $1 \leq m \leq \infty$, indicates the tolerance of the required clustering. It determines the level of overlapping between clusters. The higher the fuzziness coefficient is, a larger number of data objects will be categorized into a fuzzy band and carry a membership value between 0 and 1. We use $m = 2$ in this application.

The M-step will recalculate the centroids, i.e. the new centres of clusters, in each iteration using Equation 4.11 (Han et al., 2011).

$$c_j = \frac{\sum_{i=1}^n x_i \cdot w_{ij}^2}{\sum_{i=1}^n w_{ij}^2} \quad (4.11)$$

where $j = 1, 2, \dots, k$, and w_{ij} representing the value of the degree of membership calculated using Equation 4.10 and x_i is an data object.

In each iteration of the fuzzy clustering algorithm, we aim to minimise the following sum of the squared *error*(*SSE*) defined in Equation 4.12.

$$SSE(C) = \sum_{i=1}^n \sum_{j=1}^k w_{ij}^p \text{dist}(x_i, c_j)^2 \quad (4.12)$$

where w_{ij} represents the degree of membership for an object x_i belonging to a cluster C_j and $\text{dist}(x_i, C_j)$ measures the distance between x_i and the center of the cluster C_j . Also, p ($p \geq 1$) determines the effect of the degrees of membership ($p = 1$ in this context) (Han et al., 2011).

Such an iteration process in FCM clustering completes when the cluster centres converge or the change to the degrees of membership is less than a pre-set *epsilon* threshold value. In each iteration, we use Equation 4.13 to calculate the difference, \mathcal{E} , between each pair of new and old degrees of membership in the most recent two iterations across all the data objects. In this application, we set the maximum difference between membership values in the adjacent two iterations, i.e. the termination criterion, as 0.0005.

$$\mathcal{E} = \Delta_i^n \Delta_j^k |w_{ij}^{l+1} - w_{ij}^l| \quad (4.13)$$

where, w_{ij}^l and w_{ij}^{l+1} respectively represent the degree of membership at iteration l and $l + 1$ and the operator Δ returns the largest value in a given vector of values. Therefore, in this way, a maximum change to the membership values among all the data objects can be identified. The overall FCM clustering algorithm is listed in the following.

Algorithm 4.5. Fuzzy C-means Clustering (Han et al., 2011)

Input: Inputs file containing:

- (1) The number of test data points
- (2) The pre-determined number of clusters

- (3) The number of dimensions of the data points
- (4) The fuzziness coefficient (more than 1, $m = 2$)
- (5) The pre-defined termination criterion (a threshold value, i.e. 0.0005)
- (6) The overall test data objects

Output: The membership matrix

Begin

1. Initialize parameters including the number of test data points, the pre-determined number of clusters, the number of dimensions of the data points, the fuzziness coefficient and the termination threshold value.
2. Initialize degrees of membership for all data points for each cluster with random values.

Repeat

3. Calculate the centers of cluster vectors using Equation 4.11.
4. Update the degrees of membership for all data points using Equation 4.10.
5. Identify differences between new and old memberships for all data points and find the maximum difference using Equation 4.13.

Until *the maximum difference \leq the termination threshold;*

End

In this application, we use the above-derived set of facial actions with 18 dimensions for the representation of one facial expression data as inputs to the fuzzy clustering algorithm. The pre-defined numbers of clusters are set to 8 because the aim of this application is to recognise the eight basic emotions including contempt and neutral expressions. The clustering algorithm outputs the predicted eight membership values representing each emotion cluster. In other words, each facial expression input represented by 18 facial AUs will have a membership value for each fuzzy emotion category. The CK+ database is the only database among selected databases that provide emotion annotation for each emotion category. Moreover, for the evaluation purposes, we select the one with the highest membership value from the fuzzy clustering outputs as the potential detected emotion for each test image. Since for each test instance, the sum of its membership values across all the eight emotion clusters will be 1, a threshold value ($1/8 = 0.125$) based on the equal distribution of the test data among the eight clusters is also defined as the criterion for one emotion clustering result to be counted as a valid final detected emotion output. For example, if the highest membership clustering output of a test instance is less than this threshold value, then this test image will not be regarded as carrying any of the eight existing known emotions but indicating a newly arrived novel unseen emotion category.

Thus this clustering approach allows us to recognise the arrival of a new emotion class. Since it provides a degree of membership for each emotion cluster for each test object, which indicates how close this test instance is to the centre of each emotion vector, it also enables us to identify a compound emotion and initial experiments have also been conducted to test its efficiency.

In this research, this FCM clustering for emotion recognition is implemented in C++ language. This FCM clustering is evaluated using the CK+ database for the detection of the eight basic emotions including contempt and neutral expressions. The test set of 127 FACS coded peak emotional images and 30 'neutral' images from the CK+ database used for the AU intensity estimation are employed to evaluate this clustering based emotion recognition. A majority vote is used to label each emotion cluster derived by the clustering algorithm with emotion annotations of images provided by CK+. The evaluation results show that the FCM clustering technique achieves a promising average accuracy rate of 90.38% for the detection of the eight basic emotions. Detailed results are provided in Section 4.4.

Moreover, some initial testing on compound emotion is also conducted in order to prove its recognition capability. As discussed previously, Martinez and Du (2012) proposed a new theoretical cognitive model for the description of multiple compound emotion categories such as happy surprise or angry surprise. Some compound emotional expression images shown in Figure 4.9 have been borrowed from their work in order to test the emotion clustering method. The overall system works as follows: First of all, our unsupervised facial point detector is used to generate 54 facial points for each of these images. Then AU intensity estimation is also conducted using SVRs for the selected 18 AUs. The estimated intensity outputs of the 18 AUs are then used as an input vector to the FCM clustering algorithm to detect emotion. The top two emotion clustering outputs above the pre-defined threshold value (0.125) are selected as the final emotion clustering results for each test compound expression image. The example clustering results are also provided in Figure 4.9.

As shown in Figure 4.9, the FCM clustering technique can identify two comparatively valid higher membership values for each of the first three compound facial

expression inputs while it is also able to generate a highest sole dominating membership value for the last ‘surprised’ facial image. This indicates its great potential and flexibility for tackling challenging compound emotion recognition. For example, the clustering technique produces the highest membership for the ‘surprise’ emotion category among all the eight emotion clusters respectively for both ‘happy and fearful surprised’ facial expressions. It also generates the second highest valid membership values respectively for ‘happy’ and ‘fear’ emotion clusters for these two compound facial images. The clustering algorithm also produces two very close highest membership values for the ‘disgust’ and ‘surprise’ emotion clusters for the ‘disgusted surprise’ compound expression since it seems that this image carries comparatively less surprised indication in comparison to other compound surprised test images but also containing clear physical cues for disgusted expressions. The last example image with strong physical cues of a surprised expression is with a sole dominating membership value for the ‘surprise’ emotion cluster.



Figure 4.9. Example compound surprise facial expressions discussed in Martinez and Du (2012) and their corresponding emotion clustering results (from left to right, including happy surprise, fearful surprise, disgusted surprise, and surprise).

The above results also indicate the efficiency of the facial point detection and AU intensity estimation presented in this research, which lay a solid foundation for the clustering based compound emotion recognition. For instance, besides the deriving of AU1, 2, 5, 25, 26/27 which indicate a dominating surprised facial expression, the SVR-based AU intensity estimation also respectively identifies intensities of AU6 and AU12 for the ‘happy surprise’ compound expression and AU20 for the ‘fearful surprise’ facial image. Regression-based AU intensity estimation also produces intensities of AU1, 2, 4, 7, 17, 23 and 24 for the ‘disgusted surprise’ facial image. In future work, further evaluation will be conducted for this clustering based emotion recognition by employing databases with compound and more basic emotion annotations to further prove its robustness.

4.4 Evaluation

The overall system including facial landmark point detection, regression-based AU intensity estimation and emotion clustering have been implemented using C++ under Ubuntu. The OpenCV window on the computer is used to output the real-time video with the real-time facial landmark extraction and tracking. The AU and emotion recognition results are displayed to the user using two modes such as text messages shown on the computer terminal and speech synthesised by a text-to-speech engine.

During the testing, a database image is displayed in front of the camera or the user is required to pose a specific emotional facial expression in front of the camera. When an emotion is detected for each posed or spontaneous facial expression, the system conducts speech-based interaction to communicate back about the details of the facial emotion recognition results. Its speech synthesis engine is therefore activated to report the features of the upper and lower facial parts of an emotional facial image and also inform the user the emotion embedded in the real-time input facial expression. We present the testing conducted in detail in the following.

4.4.1 Evaluation of facial landmark generation

We have created different test sets containing 200 images from each of the five selected databases: CK+, LFW, PUT, LFPW and Helen to evaluate our facial feature point detection. As discussed earlier, CK+ contains 593 sequences of posed frontal face images contributed by 123 subjects. The PUT database (Kasiski et al., 2008) is especially designed to provide a benchmark dataset with a significant size and diversity to evaluate the efficiency and robustness of face pose estimation, 2D/3D statistical face shape model development, and pose invariant face recognition algorithms. The database consists of 9971 images from 100 subjects with 30 landmark annotation points provided for each image. Images in PUT show pose variations, partially controlled illumination conditions, occlusions or structural disturbances (e.g. make-up, glasses, facial hair, etc.) with a uniform background. The LFW database (Huang et al., 2007) has images captured from real-life situations with spontaneous facial expressions. The images in this database usually show different head poses and facial expressions, partial occlusions and background noise, which

pose great challenges to automatic facial analysis and emotion recognition systems. LFW contains more than 13,000 images of faces collected from the web with each image annotated with 20 facial points. The LFPW database (Belhumeur et al., 2011) provides 1432 real-life face images obtained via simple text queries through several search engines. Each image was originally labelled with 29 landmark points whereas a training set of 811 images from LFPW has been re-annotated with 68 landmarks by Sagonas et al. (2013). The Helen database (Le et al., 2012) is also composed of real-world high-resolution fully annotated face images. It contains 2000 training and 330 test images. Each image is originally annotated with 194 facial points. The training set of the 2000 images has also been re-annotated with 68 landmarks by Sagonas et al. (2013). We select these five databases as benchmark datasets for system evaluation since they provide diverse posed and spontaneous emotional facial expressions with frontal and multi-views, pose variations, scaling differences, partial occlusions and background clutter. As mentioned earlier, the five databases also provide landmark annotations to allow for the evaluation of the proposed facial point detector.

The existing well-known supervised learning algorithms such as AAM, CLM (Cristinacce & Cootes, 2006) and GN-DPMs (Tzimiropoulos & Pantic, 2014) are also compared with the proposed unsupervised facial point detector. The GN-DPM implementation is provided by the authors (Tzimiropoulos & Pantic, 2014) on their website, which specifies that it can detect 49 landmark points, it has been trained using 811 images from LFPW and 2000 images from Helen with the publicly available 68-point landmark configurations provided by Sagonas et al. (2013). However, the AAM and CLM models have been re-trained using the same training set of GN-DPM because the training code for GN-DPM model is not released by the authors. The three models, i.e. AAM, CLM and GN-DPMs, are used to perform landmark detection using the same 1000 images as those used for the evaluation of this research with 200 selected from each of the above five databases.

We calculate landmark detection accuracy in order to compare the performances of AAM, CLM, GN-DPMs and our model. An average error is calculated in the following way for a whole set of detected landmarks for each model. For example, if we consider

$f(x1_n, y1_n)$ as an original set of landmarks provided by the database and $f(x2_n, y2_n)$ as the set of landmarks generated after applying a feature point detector, where $n = \{1, 2, \dots, N\}$ is the number of landmarks, we calculate the absolute value of the difference between this pair of facial points using the following formula: $err_n = |x1_n - x2_n| + |y1_n - y2_n|$, where ‘|’ represents the mathematical absolute symbol which only returns positive difference values. Subsequently the average error for a whole set of landmarks is calculated below:

$$img(err) = \frac{err_0 + err_1 \dots + err_n}{n} \quad (4.14)$$

where err_n represents the error calculated for each derived landmark and $img(err)$ is the total image error for a whole set of generated landmarks. We then obtain the landmark detection accuracy rate by the following formula: $1 - img(err)$, which has been used to contribute to the results shown in Table 4.1.

Table 4.1. Evaluation results of the proposed facial point detector, AAM, CLM and GN-DPMs for landmark detection

| | CK+ (%) | LFW (%) | PUT (%) | LFPW (%) | HELEN (%) | Average (%) |
|------------------|---------|---------|---------|----------|-----------|-------------|
| AAM | 69 | 64 | 60 | 71 | 72 | 67 |
| CLM | 71 | 70 | 65 | 75 | 75 | 71 |
| GN-DPM | 80 | 75 | 75 | 82 | 85 | 79 |
| Our Model | 80 | 73 | 78 | 85 | 85 | 80 |

We notice that each of the four selected models including our model, AAM, CLM and GN-DPMs, generate a different number of landmark outputs with 54 landmarks for our model, 68 landmarks respectively for AAM and CLM and 49 for GN-DPMs. In order to conduct a fair evaluation, we use 49 landmark points for the evaluation of all the models. For example, we discard 14 points for the description of the facial contour respectively produced by AAM and CLM to obtain a set of 54 landmarks for both AAM and CLM. Then we further take off 3 landmark points for the chin and 2 landmark points for the inner corners of the mouth from the remaining landmarks generated by AAM and CLM and the 54 landmarks generated by our model to retrieve a set of 49 landmarks. Thus the average error for a set of detected 49 landmarks is calculated for AAM, CLM, GN-DPM and our model respectively for comparison.

Also, the number of landmark annotation provided by LFW and PUT database is different from other databases. For example, the CK+, Helen and LFPW databases provide 68 landmarks for each image while LFW offers 20 landmarks, and PUT provides 30. In order to conduct effective evaluation across databases, we also discard the corresponding 17 landmarks for the facial contour description and 2 points for the inner corners of the mouth among the 68 annotation points respectively provided by the CK+, Helen and LFPW databases and employ the remaining 49 points as the ground truth for the evaluation of AAM, CLM, GN-DPM and our facial point detector. Also, since the LFW and PUT databases respectively provide only 20 and 30 landmarks for the annotation of each facial image, we use the direct neighbourhood in the range of e.g. 10-13 pixels of each original landmark annotation to search if there is any generated landmark present. If there is no generated landmark within the region, then this corresponding original landmark will not be considered for evaluation. Experiments with the direct neighbourhood checking of the generated landmarks in the range of other number of pixels have also been conducted which also draw similar evaluation conclusions. Using this method, we are also able to evaluate the four facial point detectors using landmark annotations provided by LFW and PUT.

Table 4.1 illustrates the landmark detection results of our model, AAM, CLM and GN-DPM with 200 test images from each of the five selected databases. Overall, our proposed unsupervised method has outperformed these existing supervised models and achieved an average accuracy rate of 80% for CK+, 73% for LFW, 78% for PUT, 85% for both LFPW and Helen without any prior training required. As shown in Table 4.1, our unsupervised model shows robust and efficient performance when tested with images from the five selected databases with great diversity. Also, the results show that AAM and CLM models performance dropped when tested with both LFW and PUT databases even though they were trained using other databases (i.e. LFPW and Helen) and did not possess any knowledge of these two new test datasets in their learned models, which also contain diverse challenging cases such as pose variations, occlusions and scaling differences, etc. However, both AAM and CLM seem to adapt better to the testing of CK+ which contains less challenging purely posed frontal facial expressions. Therefore, the performances of

AAM and CLM have dropped dramatically when tested with PUT and LFW in comparison to those obtained from the testing of LFPW, Helen and CK+. Among all the selected supervised models the GN-DPM model achieves competitive performance in comparison to our facial point detector. The GN-DPM achieves highest accuracy rate when tested with HELEN and LFPW followed by the results for CK+, PUT and LFW. The competitive performance of GN-DPM is mainly because of the employment of Gauss-Newton optimisation to minimise a joint cost function of the global shape and local appearance models to generate a joint translational motion model. In comparison to GN-DPM, AAM and CLM, our proposed unsupervised landmark detector does not require any prior knowledge of any of the test databases and shows great flexibility and robustness in dealing with facial point detection tasks against head rotations, pose variations, scaling differences, occlusions and background clutter for images across databases. Figure 4.10 shows example landmark detection outputs of AAM, CLM, GN-DPM and our model for example images extracted from the three new test databases, CK+, LFW and PUT with images selected from LFW and PUT containing rotations and/or occlusions.

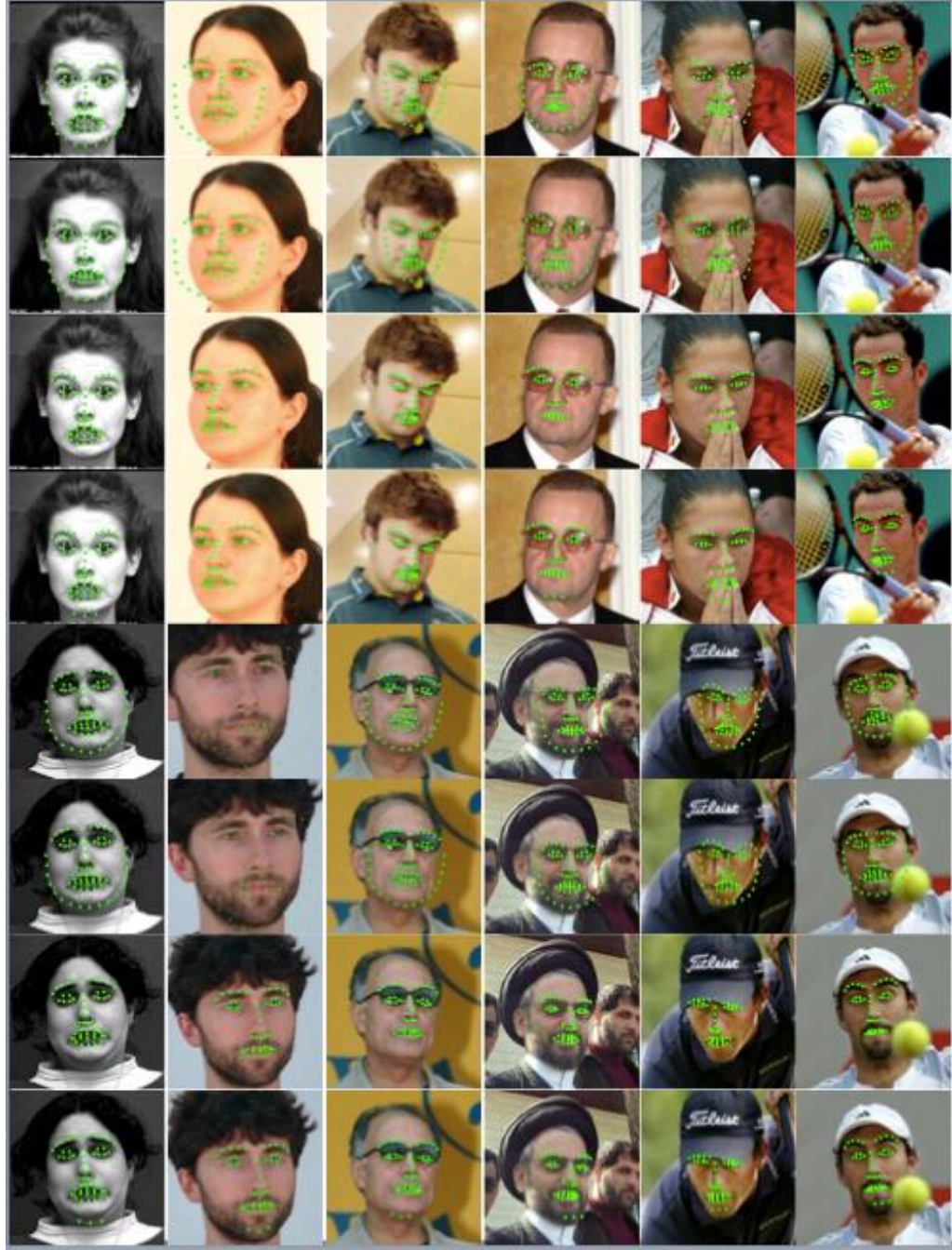


Figure 4.10. Example facial point detection outputs using images from CK+ (the first image), PUT (the second image) and LFW (the last four images). For each of the two sets of images, the first-row results generated by AAM, the second row generated by CLM, the third row generated by GN-DPM and the last row generated by our model.

4.4.2 Evaluation of landmarks generation for images with occlusion

As discussed earlier, the images with the occluded facial region will be processed using FCM and related post processing in order to further adjust the shape of the

reconstructed neutral facial element by ICP. However, in the above experiments shown in Table 4.1, only a few images from LFW, LFPW and Helen contain occlusions, CK+ provides purely frontal-view emotional images and PUT containing test images mainly with rotations. Moreover, the occlusion images from LFW, LFPW and Helen only show neutral emotion or minor emotional expression in real-life situations. Therefore, the detection accuracy for landmark generation before and after applying FCM and the related post processing for our model does not show much difference for LFW, LFPW and Helen in the above experiments shown in Table 4.1. In order to further evaluate the efficiency of FCM and related post processing for landmark generation for images with occlusions, the following experiments are conducted. First, the randomly selected 100 peak emotional images from the CK+ are edited using Adobe Photoshop in order to insert occlusion effects. For example, a black box is used to cover one of the facial elements to generate occlusions. Then another 300 images bearing either upper or lower facial occlusions are also selected from LFW, LFPW and Helen with 100 images extracted from each database for further system evaluation.

Table 4.2. Evaluation results for landmark generation for images with occlusions

| | CK+ (%) | LFW (%) | LFPW (%) | Helen (%) | Average (%) |
|------------------|----------------|----------------|-----------------|------------------|--------------------|
| AAM | 50 | 60 | 68 | 65 | 61 |
| CLM | 55 | 64 | 73 | 75 | 67 |
| GN-DPM | 70 | 80 | 80 | 85 | 79 |
| Our model | 80 | 78 | 80 | 82 | 80 |

We have also compared the landmark detection accuracy of our facial point detector with those obtained from AAM, CLM and GN- DPM for these selected 400 occluded test images. Table 4.2 shows the detailed evaluation results. On average, our proposed unsupervised approach has outperformed the three existing supervised models of AAM, CLM and GN-DPM and achieved an accuracy of 80% for images with occlusions from CK+, 78% for LFW, 80% for LFPW and 82% for Helen. Again in comparison to our model, the GN-DPM model achieves competitive performance with slightly better accuracy for face alignment for LFW and Helen databases, a similar performance of face alignment for LFPW and lower accuracy for landmark detection for CK+. AAM and CLM show limitations to such challenging face alignment tasks and achieve comparatively worse

performances especially for LFW and CK+ in comparison to GN-DPM and the proposed approach.

Furthermore, a detailed inspection of our approach also indicates that when only ICP is applied, the detection process of our model tends to suffer from poor fitting accuracy for emotional images from CK+ since ICP is only able to reconstruct a set of neutral landmarks and may not be able to restore the best-fitted geometry for occluded emotional facial components. However, after applying FCM and related post correlation processing to adjust the neutral facial landmarks further to recover the emotional indication of the occluded facial element with the knowledge of non-occluded facial regions as inputs, the detection accuracy of our model has been greatly improved, especially for emotional images from CK+.

Table 4.3. Landmark detection accuracies for upper and lower facial occlusions (UO: upper facial occlusion. LO: lower facial occlusion).

| Upper/lower occlusions | UO | UO | UO | UO | LO | LO | LO | LO |
|------------------------|---------|---------|----------|-----------|---------|---------|----------|-----------|
| Dataset | CK+ (%) | LFW (%) | LFPW (%) | Helen (%) | CK+ (%) | LFW (%) | LFPW (%) | Helen (%) |
| AAM | 50 | 60 | 70 | 70 | 45 | 50 | 60 | 60 |
| CLM | 55 | 65 | 80 | 75 | 50 | 55 | 60 | 70 |
| GN-DPM | 80 | 90 | 85 | 85 | 60 | 70 | 75 | 75 |
| Our model | 90 | 80 | 85 | 80 | 70 | 70 | 70 | 75 |

In order to further identify the efficiency of the proposed facial point detector, we also divide the occluded images into upper and lower facial occlusion categories and provide the detection accuracies for these two categories respectively for the four facial point detection models in Table 4.3. In order to obtain a fair split between upper and lower facial occlusion cases across databases, we select 40 images from the CK+ database and transform half of them into upper facial occlusions and the other half of them into lower facial occlusions using Adobe Photoshop whereas a set of 40 images is also extracted respectively from LFW, LFPW and Helen with 50% indicating upper facial occlusions and 50% showing lower facial occlusions (i.e. a total of 120 images are extracted from LFW, LFPW and Helen). The results shown in Table 4.3 indicate that overall landmark detection achieves better accuracy for the upper facial occlusions than for lower facial occlusions irrespective of the model applied. Also, GN-DPM (with an average accuracy of 77.5%)

and our model (with an average accuracy of 77.5%) achieve comparably better detection accuracies for both upper and lower facial occlusions than AAM (with an average accuracy of 58.13%) and CLM (with an average accuracy of 63.75%). GN-DPM outperforms our model when LFW is used whereas our model outperforms GN-DPM when the transformed occluded images from the CK+ database are used. Overall, our model achieves a slightly lower average accuracy of 83.75% compared to GN-DPM with an average accuracy of 85% for landmark detection for upper facial occlusions and a slightly higher average accuracy of 71.25% than GN-DPM with an average accuracy of 70% for landmark detection for lower facial occlusions. Figure 4.11 shows example landmark detection results for images with occlusions from the CK+ database.



Figure 4.11. Landmark detection for images with occlusions from the CK+ database.

4.4.3 Evaluation of landmarks generation for images with rotations

Since Gabor filter and BRISK are rotation invariant facial feature detectors, the proposed feature detection model of this research shows great capacity in dealing with images with rotations and pose variations. After applying Gabor filter + BRISK + ICP, it is able to produce 54 landmarks for each test image. Evaluated with total 400 randomly selected pose variations images from LFW, PUT, LFPW and Helen with 100 images from each database, our model outperforms all the selected models by achieving 81% landmark detection accuracy. Table 4.4 illustrates the detailed evaluation results for images with rotations. Again, the GN-DPM model has outperformed our approach when tested with LFW whereas our model performs better than GN-DPM for PUT and Helen databases for rotated images. Overall, without any prior knowledge of the application domain or training conducted with images with occlusions or rotations, the proposed unsupervised facial point detector shows impressive adaptability and robustness, achieves competitive/highest

average accuracies and outperforms the three supervised models for majority cases in the experiments conducted.

Table 4.4. Evaluation results for landmark generation for images with rotations.

| | LFW (%) | PUT (%) | LFPW (%) | Helen (%) | Average (%) |
|------------------|---------|---------|----------|-----------|-------------|
| AAM | 65 | 60 | 70 | 70 | 66 |
| CLM | 70 | 65 | 70 | 75 | 70 |
| GN-DPM | 80 | 70 | 80 | 80 | 77 |
| Our model | 75 | 85 | 80 | 85 | 81 |

Our facial point detector also shows optimal computational costs during the evaluation. Based on the experiments conducted across the five databases, the averaged processing time of one facial image for our facial point detector is 80 ms, respectively 124 and 59 ms faster than the performance of AAM and CLM. Our model also shows slightly better computational efficiency than GN-DPM. The detailed average processing time of one facial image for the four models is listed in Table 4.5.

Table 4.5. The average processing time of a single facial image for AAM, CLM, GN-DPM and our model.

| Models | Average processing time per image (ms) |
|------------------|--|
| AAM | 204 |
| CLM | 139 |
| GN-DPM | 83 |
| Our model | 80 |

4.4.4 Evaluation of AU intensity estimation and emotion recognition

For AU intensity estimation, as discussed earlier, we have employed 18 SVRs and 18 NNs to measure the intensities of the 18 selected AUs respectively. The training set of 200 FACS coded peak emotional images of the seven emotions and 50 neutral images obtained from the CK+ database is used respectively for the training of both of the 18 SVRs and 18 NNs. The AU annotations for neutral images are not provided by CK+ database. In order to detect neutral emotion AUs, we consider that all the AUs to have ‘0’ intensity value. Then 127 images for the seven emotions and 30 images from the neutral category from the CK+ database are used to evaluate the performances of SVR and NN-based AU intensity regression. We also automatically extract 54 landmarks for each of these test

images using our facial point detector. The landmark points of corresponding upper and lower facial elements are then used as inputs to the SVR and NN-based AU intensity estimation. MSEs of the intensity estimation for the 18 AUs are also calculated for SVRs and NNs based on the comparison with the ground truth annotations provided by the CK+ database. The detailed results of MSE for both SVRs and NNs are provided in Table 4.6. (The last column in Table 4.6 also shows the MSE differences between each pair of SVR and NN.)

Table 4.6. Mean squared errors of AU intensity estimation for SVRs and NNs.

| Facial AUs | MSE for SVRs | MSE for NNs (in comparison to SVRs) |
|------------|-----------------|--|
| AU1 | 0.05324 | +0.0026 |
| AU2 | 0.05445 | +0.0211 |
| AU4 | 0.02934 | +0.0049 |
| AU5 | 0.02951 | +0.0010 |
| AU6 | 0.02800 | +0.00155 |
| AU7 | 0.02925 | -0.00100 |
| AU9 | 0.062113 | -0.00040 |
| AU10 | 0.00035 | +0.00003 |
| AU12 | 0.02740 | +0.00153 |
| AU14 | 0.00070 | +0.00021 |
| AU15 | 0.00051 | +0.00055 |
| AU16 | 0.04721 | -0.00122 |
| AU17 | 0.09429 | +0.0041 |
| AU20 | 0.00065 | +0.00001 |
| AU23 | 0.06300 | +0.00200 |
| AU24 | 0.05999 | -0.00025 |
| AU25 | 0.08302 | +0.00365 |
| AU26/27 | 0.05223 | +0.00144 |

As shown in Table 4.6, the SVR-based intensity measurement outperformed the NN-based method. In comparison to the NNs, the SVR-based approach performs well for the intensity measurement for the following AUs: AU1, 2, 4, 5, 6, 10, 12, 14, 15, 17, 20, 23, 25, and 26/27, while it needs improvements for the intensity measure for AU7, 9, 16 and

24. For both SVR and NN-based regression, MSE for the following AUs is less than 0.05: AU4, 5, 6, 7, 10, 12, 14, 15, 16, and 20 with AU17 posing the maximum MSE for both approaches. We also produce the evaluation results based on the ‘presence’ or ‘absence’ of each AU based on the comparison of the results produced by SVRs and NNs and the original database annotations. The SVRs show great performance (more than 80% detection accuracy) for the detection of the following AUs: AU1, 2, 4, 5, 9, 12, 14, 17, 25, 26/27, while the NNs perform well for the identification of AU1, 4, 5, 12, 17, 25, and 26/27.

The estimated intensities of the 18 AUs for each of the 157 (127 emotional + 30 neutral) test images from CK+ are then used to detect the seven basic emotions and neutral expressions. The evaluation results of the fuzzy clustering based emotion recognition are provided in Table 4.7. Generally, each emotion category is reasonably recognised. Among the eight emotions, surprise and happiness are reliably detected respectively with 98% and 97% accuracies followed by well-detected disgust with 95% and neutral with 93% accuracies. Sadness, anger and fear are reasonably recognised respectively with an accuracy of 90%, 85% and 85%, while contempt is with the least recognition accuracy of 80%.

Table 4.7. Emotion clustering results for the seven emotions and neutral expressions.

| | Surprise | Fear | Anger | Disgust | Sadness | Happiness | Contempt | Neutral |
|------------------|----------|------|-------|---------|---------|-----------|----------|---------|
| | (%) | (%) | (%) | (%) | (%) | (%) | (%) | (%) |
| Surprise | 98 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| Fear | 0 | 85 | 6 | 5 | 4 | 0 | 0 | 0 |
| Anger | 0 | 0 | 85 | 10 | 5 | 0 | 0 | 0 |
| Disgust | 0 | 2 | 0 | 95 | 3 | 0 | 0 | 0 |
| Sadness | 0 | 0 | 5 | 3 | 90 | 0 | 0 | 2 |
| Happiness | 0 | 0 | 0 | 1 | 0 | 97 | 0 | 2 |
| Contempt | 0 | 10 | 0 | 10 | 0 | 0 | 80 | 0 |
| Neutral | 0 | 0 | 2 | 0 | 5 | 0 | 0 | 93 |

We have also compared our results with other well-known image based emotion detection applications. For example, Shan et al. (2009) employed Boosted-LBP based SVM to recognise six basic emotions and neutral with the evaluation conducted using

images from CK+. Jain et al. (2011) presented a facial emotion recognition system from video sequences using Latent-Dynamic Conditional Random Fields. Their model performed temporal modelling of shapes with the evaluation conducted using image sequences. Wu et al. (2010) employed spatiotemporal Gabor filters for automatic facial emotion recognition. Because of the above three related developments' state-of-the-art performance and the employment of the same database (CK+) for evaluation, we select these applications to compare with our research to further evaluate our system's efficiency. Moreover, our previous research (Zhang et al., 2013) employed the 31 facial points generated by NAO robot's vision APIs to detect 17 AUs and the six basic emotions from CK+ database images. We also employ this previous development for system comparison in order to prove the efficiency of the new development. Therefore, Table 4.8 shows the comparison between the above four related applications and this research.

Generally, our system outperforms all the above related research, i.e. spatiotemporal Gabor filtering based facial emotion recognition (Wu et al., 2010), the approach of Latent-Dynamic Conditional Random Fields (Jain et al., 2011) and the Boosted-LBP based SVM (Shan et al., 2009), especially for the recognition of anger, sadness and neutral. In comparison with the Boosted-LBP based SVM (Shan et al., 2009), our research focuses on eight-class emotion recognition and shows comparatively more stable performance for the recognition of all emotion categories. It also outperforms our previous work (Zhang et al., 2013) significantly, which was conducted purely based on the original vision APIs of the robot. The above comparisons also further indicate the efficiency of the proposed facial point detector, regression-based AU intensity estimation and the clustering based emotion recognition.

Table 4.8. Comparison with other related work.

| | Zhang et al. (2013) (%) | Wu et al. (2010) (%) | Jain et al. (2011) (%) | Shan et al. (2009) (%) | This Work (%) |
|------------------|----------------------------|-------------------------|---------------------------|---------------------------|------------------|
| Surprise | 77 | 87.9 | 99.06 | 97.3 | 98 |
| Fear | 65 | 66.7 | 94.37 | 79.9 | 85 |
| Anger | 90 | 82.9 | 76.71 | 85.1 | 85 |
| Disgust | 83 | 67.7 | 81.51 | 97.5 | 95 |
| Sadness | 60 | 78.4 | 77.22 | 74.7 | 90 |
| Happiness | 80 | 87.7 | 98.55 | 97.5 | 97 |
| Contempt | - | - | - | - | 80 |
| Neutral | - | - | 73.46 | 92 | 93 |
| Average | 75.83 | 78.6(7) | 85.84(7) | 89.14(7) | 90.38(8) |

Although we discuss the employment of image databases for the evaluation of each core function of this research to prove its efficiency, the system is also able to perform real-time facial point detection and emotion recognition from real testing subjects with pose variations (at least up to 60 degrees), scaling differences, and partial occlusions, etc. Especially 30 posed compound facial expression data are gathered from five real subjects aged between 25 and 35 with each subject contributing five images respectively for the following five categories of compound surprised expressions: happy surprised, angry surprise, fearful surprise, sad surprise, disgusted surprise and surprise. Evaluation results show that pure surprise emotional expressions achieve the highest 90% recognition accuracy followed by the happy surprise of 88% detection accuracy, fearful surprise with 83%, angry surprise with 81% and 80% accuracy for a disgusted surprise. Sad surprise has the lowest result of 75%. We notice that some compound emotions, e.g. happy/angry/fearful surprised expressions, tend to have comparatively stronger physical cues than other compound emotions such as disgusted/sad surprise emotions. Therefore, happy/angry/fearful surprised emotions are better recognised than disgusted/sad surprise indications. In future work, evaluation with real testing subjects will be provided to further indicate the system's efficiency.

4.4 Summary

In this chapter, we have developed a novel unsupervised facial point detector (a rarely explored topic), regression-based AU intensity estimation and emotion clustering for the

recognition of the eight basic and compound emotions from posed and spontaneous facial expressions. The proposed facial point detection model can perform robust landmark extraction from images with illumination changes, head rotations, pose variations, scaling differences, partial occlusions and background clutter. It also has optimal computational cost and is significantly faster than AAM and CLM with comparable computational costs to GN-DPM. Moreover, the AU intensity estimation and emotion clustering are also evaluated using images from the CK+ database. The SVR-based AU intensity estimation outperformed the NN-based method. In comparison to NNs, the SVR-based method performs well for the intensity measurement for the following AUs: AU1, 2, 4, 5, 6, 10, 12, 14, 15, 17, 20, 23, 25, and 26/27. Moreover, FCM clustering also not only enables the system to recognise the seven basic emotions and neutral expressions but also shows great potential to detect compound and newly arrived novel emotion classes. It also outperforms other state-of-the-art research in the field.

Chapter 5 A micro-GA Embedded PSO Feature Selection Approach to Intelligent Facial Emotion Recognition

In Chapters 3 and 4, we discussed the shape and texture-based feature extraction. The proposed model in Chapter 3 employs both shape and texture based features, however, the extracted features have very high-dimensions which make it computationally intensive. Although the unsupervised facial point detector model proposed in Chapter 4 is computationally efficient, it is limited to the geometric information of the face. Therefore, in order to address the drawbacks in the previously proposed models, we propose a facial expression recognition system using evolutionary micro-GA embedded PSO-based feature optimisation. The system first employs modified LBP, which conducts horizontal and vertical neighbourhood pixel comparison, to generate a discriminative initial facial representation. Then, a PSO variant embedded with the concept of mGA, i.e. mGA-embedded PSO, is proposed to perform feature optimisation. It incorporates a non-replaceable memory, a small-population secondary swarm, a new velocity updating strategy, a sub-dimension based in-depth local facial feature search, and a cooperation of local exploitation and global exploration search mechanism to mitigate the premature convergence problem of conventional PSO. Multiple classifiers are used for recognising seven facial expressions.

5.1 The Proposed Facial Expression Recognition System

This section presents the detailed methodology of the proposed facial expression recognition system. The proposed system employs a novel variant of LBP algorithm for facial feature extraction and a novel improved variant of PSO for selecting the most discriminative features. Figure 5.1 illustrates the system architecture. The detailed explanation is presented as follows.

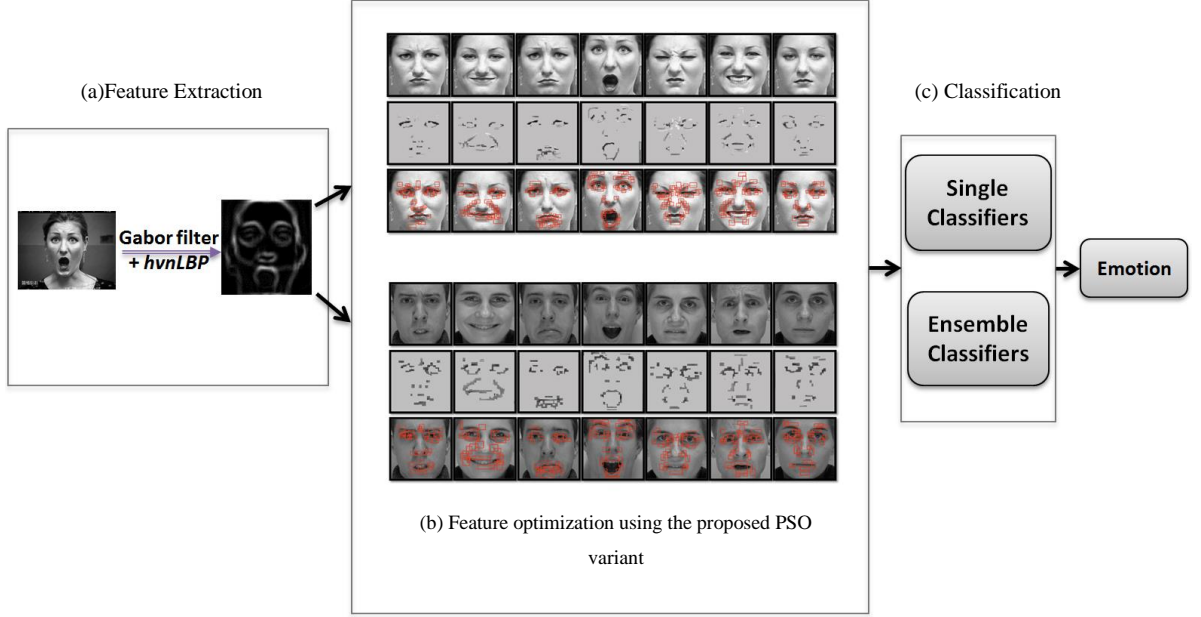


Figure 5.1. The system architecture

5.1.1 Facial feature extraction using the proposed LBP model

In this research, in order to improve the discriminative abilities of LBP, we propose horizontal and vertical neighbourhood pixel comparison LBP (*hvnLBP*), which is motivated by the limitations of conventional LBP (see below). The proposed LBP is also integrated with the Gabor filter for producing discriminative facial representations.

There are four steps in the feature extraction process: pre-processing for illumination changes and noise invariance, face detection, Gabor magnitude image generation, and the proposed *hvnLBP* based textural description. First of all, we apply histogram equalization and bilateral filter to compensate illumination variations and reduce noise in the input image, respectively. We then use a Haar-cascade face detector to detect faces. A 2D Gabor filter is also applied to produce magnitude pictures. Finally, the proposed *hvnLBP* operator is used to generate the textural description of facial images.

Conventional LBP (Ojala et al., 1996) employs a circular neighbourhood for feature extraction. This original LBP operator performs a comparison purely between the central pixel and the eight surrounding neighbourhood pixels and therefore is likely to lose the contrast information among the neighbourhood pixels. To solve this problem, we propose *hvnLBP* to capture missing contrast information among the neighbourhood pixels. Instead

of comparing with the central pixel as in original LBP, *hvnLBP* employs horizontal and vertical neighbourhood pixels for direct comparison to produce the resulting textural descriptions. As an example, we employ $P = \{P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7\}$ to represent the eight neighbourhood pixels in LBP, as shown in Figure 5.2. In either vertical or horizontal comparison, the values of the vertical or horizontal neighbouring pixels are compared with one another. A ‘1’ is assigned to the pixel with the highest value and a ‘0’ is assigned to the remaining pixels. This horizontal and vertical comparison process can be conducted in any order, i.e. horizontal comparison followed by vertical comparison, or vice versa. Moreover, in both vertical and horizontal comparisons, we do not include the centre pixel for comparison. Referring to Figure 5.2, as an example, for horizontal comparison, we first compare the pixel sets of $\{P_0, P_1, P_2\}$, $\{P_7, P_3\}$, and $\{P_6, P_5, P_4\}$. Subsequently, we conduct the vertical comparison with the pixel sets of $\{P_0, P_7, P_6\}$, $\{P_1, P_5\}$, and $\{P_2, P_3, P_4\}$. If a pixel has conflicting outputs in the horizontal and vertical comparisons (e.g. the highest value in the horizontal comparison but not in the vertical comparison, or vice versa), then the highest value (i.e. 1) is used as the final output, since the pixel is regarded as important, which contains valuable contrast information in the dimension that generates the highest value. The mathematical representation of this proposed *hvnLBP*_{*p,r*} operator is illustrated as follows.

$$\begin{aligned} hvnLBP_{p,r} = \{ & S(\max(l_0, l_1, l_2)), S(\max(l_7, l_3)), S(\max(l_6, l_5, l_4)), \\ & S(\max(l_0, l_7, l_6)), S(\max(l_1, l_5)), S(\max(l_2, l_3, l_4)) \} \end{aligned} \quad (5.1)$$

where p is the number of neighbourhood pixels, and r is the radius. l_i represents the i^{th} neighbourhood of pixel l while S denotes the comparison operation, as follows.

$$S(\max(l_j, l_k, l_m)) = \begin{cases} 1 & \text{if maximum} \\ 2 & \text{if not maximum} \end{cases} \quad (5.2)$$

where l_j, l_k and l_m represent the neighbourhood pixels in a row or column. Note that l_k is removed if it is the centre pixel.

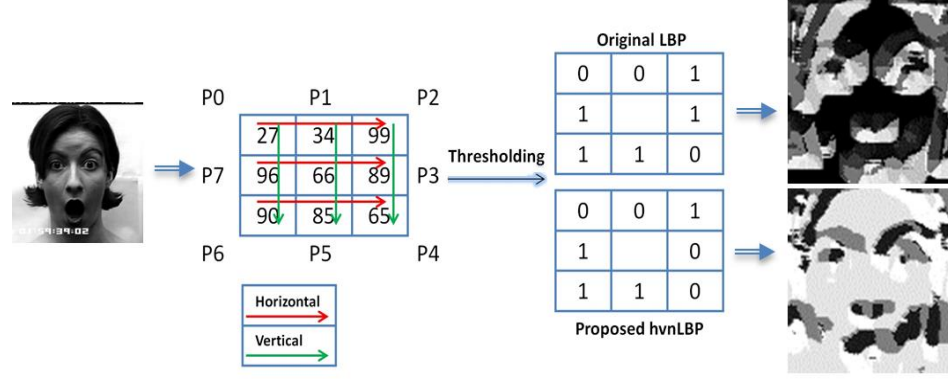


Figure 5.2. An example output of the proposed *hvnLBP* operator in comparison with that of the original LBP

An example output of the proposed $hvnLBP_{p,r}$ operator is provided in Figure 5.2, where $p = 8$ and $r = 1$. In this research, we use a window size of 75x75 pixels to represent a detected face image. Therefore, by applying the proposed *hvnLBP* operator, we obtain 25x25 (i.e. 625) sub-regions with the size of each sub-region being 3x3.

Overall, in comparison with the original LBP operator, the experimental results indicate that *hvnLBP* is more capable of capturing discriminative contrast information such as corners and edges among neighbourhoods to inform subsequent PSO-based feature selection and facial expression analysis.

5.1.2 The proposed mGA-embedded particle swarm optimisation for feature selection

To identify the discriminative characteristics of each expression, we propose a PSO variant embedded with the concept of mGA for feature optimisation, called the mGA-embedded PSO algorithm. This proposed PSO algorithm mitigates the premature convergence problem of conventional PSO and shows superior capabilities of discriminative feature selection. The proposed mGA-embedded PSO algorithm employs personal average experience and Gaussian mutation for velocity updating. Furthermore, it integrates the diversity maintenance strategy of mGA to keep the original swarm in a non-replaceable memory, which remains intact during the lifecycle of the algorithm to increase swarm diversity. Inherited from the concept of mGA, a secondary swarm with a

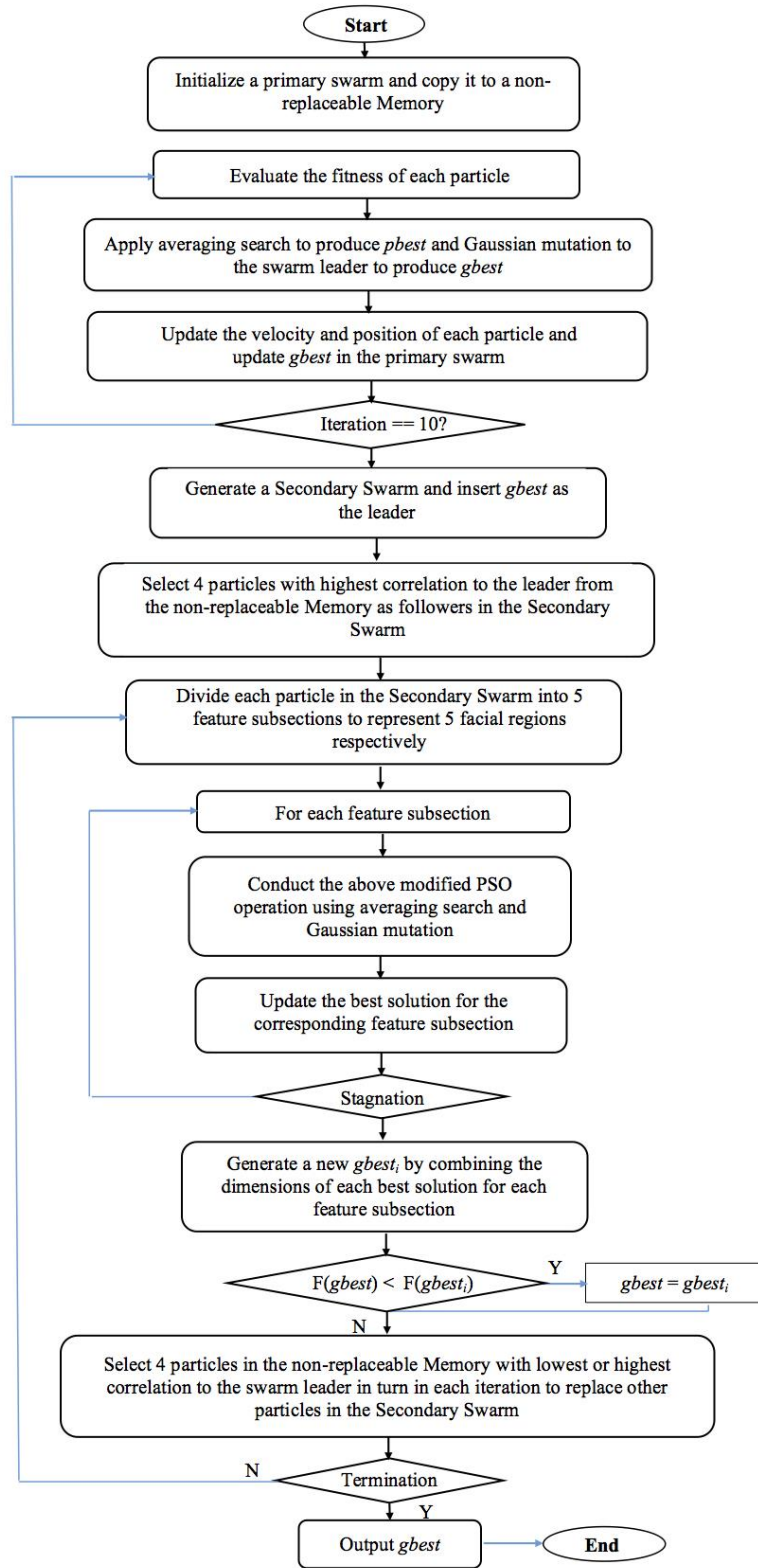


Figure 5.3. The flowchart of the proposed PSO algorithm

the small population size of five particles is employed. The swarm comprises a leader and four follower particles with the highest or lowest correlation to the leader from the non-replaceable memory to increase local and global search capabilities and avoid premature convergence. Moreover, the algorithm separates facial features into specific areas for in-depth local sub-dimension based search. Overall, the local exploitation and global exploration search strategies of the algorithm work cooperatively to lead the search process to the global optima. Algorithm 5.1 illustrates the pseudo code of the proposed mGA-embedded PSO algorithm, while Figure 5.3 shows the flowchart of the algorithm.

Algorithm 5.1: Pseudo-Code of mGA-embedded PSO

Start:

- (1) Initialize a primary swarm (e.g. 30 particles);
 - (2) Copy the initialized swarm into a non-replaceable Memory;
-

Step 1: Perform modified PSO operator

```

    For each particle in the primary swarm do
    {
        1.1 Evaluate each particle using the defined fitness function;
        1.2 Compute the average fitness value of previous runs (if available) for each particle in the
        primary swarm;
        1.3 Perform the proposed Averaging Search operation (Equation 5.6) to generate pbest for each
        individual particle;
        1.4 Apply the Gaussian mutation operation to the swarm leader to produce gbest (Equation 5.7);
        1.5 Update the velocity and position of each particle;
        1.6 Update the best particle gbest in the primary swarm;
        1.7 Until (iterations==10)
    }

```

Step 2: Generate a Secondary Swarm

```

{
    2.1 Select the best particle gbest from primary swarm as the leader of the Secondary Swarm;
    2.2 Select 4 particles that have the highest correlation with the leader from the non-replaceable Memory,
    which contains the original particle swarm, as the followers in the Secondary Swarm; //this is for local
    exploitation and a high correlation means very similar particles.
}

```

Step 3: Divide each particle in the Secondary Swarm into five feature subsections with each subsection consisting of partial dimensions which indicates a specific facial region (e.g. eye, eyebrow, mouth etc);

For each feature subsection representing each facial region do // i.e. the corresponding partial

dimensions of each particle in the Secondary Swarm

```
{  
    3.1 Apply operations of line 1.1-1.5;  
    3.2 Update the best solution for the corresponding feature subsection;  
    3.3 Until (stagnation detected);  
}
```

Step 4: Replace the particles in the Secondary Swarm

```
{  
    4.1 Combine the dimensions of each best solution for each feature subsection to replace the swarm leader  
    in the Secondary Swarm if this newly generated combined leader has a better fitness value;  
  
    4.2 Select 4 particles in the non-replaceable Memory of the original swarm that have the lowest  
    correlation with the above swarm leader to replace other particles in the Secondary Swarm; //this is for global  
    exploration, and the lowest correlation means particles with high variations to the leader  
}
```

While (Overall termination criteria are not achieved)

```
{  
    Repeat Step 3;  
    Repeat Step 4, but change from lowest correlation to highest correlation in a vice versa manner;  
}
```

End

Return the most optimal solution;

End

5.1.2.1 Updating personal best and global best

In conventional PSO, each particle represents the solution and each particle move around the search space by following the swarm leader in order to find the optimal solution. Each particle has a position in the search space represented as $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, whereas it also has a velocity represented as $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, with D denoting the dimensionality of the search space. Each particle has a memory of its best experience whose position is represented as $pbest$. The swarm leader represents the best experience of the overall swarm, whose position is represented as $gbest$. The position, x^{t+1} , and velocity, v^{t+1} , of each particle are updated using the following equations:

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (5.3)$$

$$v_{id}^{t+1} = wv_{id}^t + c_1r_1(p_{id} - x_{id}^t) + c_2r_2(p_{gd} - x_{id}^t) \quad (5.4)$$

where t and d indicate the t -th iteration and d -th dimension in the search space, respectively. An inertia weight, w , is used to embed iteration influence of the previous velocity. Note that r_1 and r_2 represent random values within the range of $[0, 1]$ whereas c_1 and c_2 are the acceleration constants. Furthermore, p_{id} and p_{gd} indicate elements of $pbest$ and $gbest$ in the d -th dimension. In this research, we modify the velocity updating Equation 5.4 by introducing the averaging search strategy for computing p_{id} and Gaussian mutation for computing p_{gd} . Specifically, the averaging search strategy takes the personal average experience into account, instead of the conventional personal best experience. The average experience is obtained by averaging the positions found from previous iterations of each individual particle for generating $pbest$. This enables the algorithm to better look into the search space in-between to increase local exploitation. Furthermore, instead of using the position of the global best experience directly, Gaussian distribution operation is applied to the swarm leader to generate $gbest$. This mutation technique enables the generation of offspring further away from its parent to increase global exploration. Therefore, the revised velocity updating strategy possesses more capability of sustaining search diversity. The updated formulas are provided below.

$$v_{id}^{t+1} = wv_{id}^t + c_1r_1(p'_{id} - x_{id}^t) + c_2r_2(p'_{gd} - x_{id}^t) \quad (5.5)$$

$$p'_{id} = \frac{\sum x_{id}}{t} \quad (5.6)$$

$$p'_{gd} = p_{gd} + (x_{max}^d - x_{min}^d) \times \phi(o, h) \quad (5.7)$$

Where p'_{id} and p'_{gd} represents the updated *pbest* and *gbest* in the d -th dimension using personal average experience and Gaussian distribution, respectively, as defined in Equations 5.6 and 5.7. Moreover, in Equation 5.7, $\phi(o, h)$ indicates the Gaussian distribution and o represents the mean of the distribution with h as the standard deviation which decreases linearly during the execution. Note that x_{max}^d and x_{min}^d indicate the upper and lower bounds of the decision vector in the d -th dimension, respectively, $d = 1, 2, \dots, D$.

As indicted in Algorithm 5.1, we first initialize the original swarm with 30 particles. The modified PSO operation with the proposed velocity updating formula is applied to the initial swarm. It iterates 10 times at the beginning of the algorithm to find the best leader. We use a small number of iterations (i.e. 10) for this initial PSO search to accelerate convergence and allow benefits from subsequent search strategies to take place. This mainly aims to find the best balance between computational costs and performance. The following setting (obtained from experimental trials) is applied to this modified PSO operation, i.e. maximum velocity = 0.6, inertia weight = 0.78, population size = 30, acceleration constant $c_1 = c_2 = 1.2$ and maximum generations = 500. Moreover, Equation 5.8 is used to define the fitness evaluation for each particle, C , which consists of two criteria, i.e. classification performance and the number of selected features. Since we apply the proposed PSO algorithm to each emotion category separately, in an attempt to identify the discriminative features for each distinct expression, the classification accuracy score in Equation 5.8 indicates accuracy of each individual expression, rather than combined accuracy across all emotion categories. This helps avoid bias towards specific emotion categories during optimization (see the related discussion in Chapter 2 Section 2.2.2).

$$fitness(C) = w_a accuracy_c + w_f (number_features_c)^{-1} \quad (5.8)$$

where w_a and w_f are two predefined weights for classification accuracy and the number of selected features, respectively, with $w_a = 1 - w_f$. In addition, parameters w_a and w_f indicate the relative importance of classification performance and the number of selected features, respectively. In this work, since the classification performance is considered to be more important than the number of selected features, w_a assumes a higher value than w_f , i.e. $w_a = 0.9$ and $w_f = 0.1$.

5.1.2.2 Constructing the secondary swarm embedded with the concept of mGA

Besides the velocity updating mechanism, the proposed PSO algorithm integrates the concepts of mGA and a secondary swarm, as well as the cooperation of local exploitation and global exploration search strategies to balance between convergence speed and swarm diversity. In summary, the proposed algorithm employs the diversity maintenance strategy of mGA using a non-replaceable memory. This non-replaceable memory comprises the initialized swarm to sustain search diversity. Motivated by the small population size concept of mGA, a secondary swarm with five particles comprising the swarm leader and four follower particles from the non-replaceable memory with the highest or lowest correlation with the leader is constructed to increase local exploitation and global exploration. A sub-dimension based search in the secondary swarm is also conducted, in order to identify the discriminative regional facial features. Moreover, the local exploitation and global exploration search strategies of the secondary swarm work in a collaborative manner to avoid stagnation and overcome premature convergence. The details of these strategies are as follows.

mGA is a small-population GA with a re-initialization mechanism. It was initially proposed by Goldberg (1989), whose theories suggested that a small population was sufficient enough to achieve convergence regardless of the chromosome length. mGA usually employs a population of 3-6 chromosomes and shows great capability of solving non-linear optimisation problems (Coello & Pulido, 2005). Instead of using the mutation operation as in classical GA, mGA employs a restart strategy to maintain genetic diversity in the population. The mGA model is proven to be more capable of avoiding premature convergence and reaching the optimal search region than the classical GA (Krishnakumar, 1990). Because of its impressive performance and fast convergence speed, mGA has been widely used to deal with single and multi-objective optimisation problems (Ali & Ramaswamy, 2009). Furthermore, Coello and Pulido (2001) proposed a multi-objective mGA with two memories, i.e. population memory and external memory. The population memory consists of replaceable and non-replaceable aspects. The non-replaceable fragment of the memory remains intact during the entire lifetime of the algorithm, in order

to bring sufficient diversity to the algorithm, whereas the replaceable portion of the memory is used for conventional evolution where the solutions are kept updated in the subsequent evolutionary cycles. The multi-objective mGA shows efficient search diversity and requires less computational cost compared with other algorithms such as Pareto Archived Evolutionary Strategy.

This research borrows the multi-objective mGA concept with the replaceable and non-replaceable memories to update the swarm leader (replaceable portion) and preserve the diversity of the initialized swarm (non-replaceable portion), respectively. After initializing the swarm with 30 randomly generated particles at the beginning of the algorithm (see Algorithm 5.1), this original swarm is stored in the non-replaceable memory, which remains intact during the lifetime of the algorithm, in order to reward swarm diversity when stagnation occurs. To balance between swarm diversity and convergence speed, a secondary swarm embedded with the small population concept of mGA is constructed. It has a typical population size of five and consists of a swarm leader and four follower particles from the non-replaceable memory. As illustrated in Algorithm 5.1, the followers are chosen based on two types of correlation relationships with the leader: (i) the lowest and (ii) the highest correlations. Particles with the lowest correlation provide higher variations in the swarm to enable global exploration whereas particles with the highest correlation bring more similarity in the swarm where local exploitation can be observed. Moreover, we define the correlation relationship between particles using Equations 5.9-5.10 (Snedecor & Cochran, 1967). Since the extracted features using *hvnLBP* are in the binary format and can be converted into histogram easily, we use the histogram correlation comparison method, as shown in Equations 5.9-5.10 (Snedecor & Cochran, 1967), to identify particles with highest/lowest correlation to the leader.

$$corr(H_1, H_2) = \frac{\sum_l (H_1(I) - H'_1)(H_2(I) - H'_2)}{\sqrt{\sum_l (H_1(I) - H'_1)^2 \sum_l (H_2(I) - H'_2)^2}} \quad (5.9)$$

where,

$$H'_k = \frac{1}{N} \sum_I H_k(I) \quad (k = 1,2) \quad (5.10)$$

where *corr* indicates the correlation relationship between two particles with H_1 and H_2 representing the histograms for the swarm leader and a follower particle, respectively. H'_k indicates the mean of the histogram for the K -th particle ($k = 1,2$), whereas N represents the number of histogram bins and I indicates the intensity range present in the histogram. Equation 5.9 produces an output in the range of $[0, 1]$, with ‘0’ and ‘1’ representing the lowest and highest correlations, respectively.

As shown in Algorithm 5.1 and Figure 5.3, first of all, after identifying the swarm leader by the previously modified PSO process, four follower particles from the non-replaceable memory with the highest correlation with the leader are recruited to the secondary swarm. The aim of extracting the follower particles from the non-replaceable memory, instead of using the particles from the main swarm, is to avoid diversity loss as the particles in the main swarm tend to be converged and become identical after 10 iterations. Moreover, these follower particles with the highest correlation with the leader provide a certain degree of position proximity in the secondary swarm, therefore enabling local exploitation of the search space. Subsequently, we divide each particle in the secondary swarm into five feature subsections, with each subsection representing each facial region to enable an in-depth local search to identify its discriminative features. This in-depth local optimal facial feature search is discussed in detail in Section 5.1.2.3. This subsection based local facial feature search reveals a new swarm leader whose fitness value is compared with that of the previous leader, in order to elect a new leader for the next iteration.

After employing particles with the highest correlation with the leader as followers to conduct an in-depth local optimal facial feature search, the secondary swarm recruits a new set of four particles with the lowest correlation with the leader from the non-replaceable memory to replace the existing follower particles. Since the new set of follower particles with the lowest correlation recruited from the original swarm inject high variation to the secondary swarm, it boosts the swarm diversity significantly to increase global exploration

and avoid premature convergence. Subsequently, the newly updated diversified secondary swarm is also used to conduct a local facial feature search (see Section 5.1.2.3) to identify a new swarm leader.

In this way, particles with the highest or lowest correlation with the swarm leader from the non-replaceable memory are recruited alternately in the secondary swarm to increase local exploitation and global exploration. Moreover, when local exploitation in the sub-dimension search using particles with the highest correlation with the leader stagnates, our PSO algorithm employs follower particles with the lowest correlation with the leader from non-replaceable memory to increase swarm diversity and drive the search out of the local optimum trap. On the other hand, when global exploration in the sub-dimension search using particles with the lowest correlation with the leader fails to generate a fitter leader, it recruits follower particles with the highest correlation to the leader from non-replaceable memory to avoid stagnation and enable local exploitation. Therefore, the local and global search mechanisms embedded in the secondary swarm work cooperatively to mitigate premature convergence and lead the search towards the global optima.

5.1.2.3 In-depth local optimal search

As discussed earlier, after particles with the highest or lowest correlation with the leader are recruited in the secondary swarm, we divide each particle in the secondary swarm into five feature subsections with each subsection consisting of partial dimensions which indicate a specific facial region (e.g. eye, eyebrow, nose, mouth, and cheek). For each facial region, we apply the above-modified PSO operation with the updated velocity updating formula defined in Section 5.1.2.1 to conduct an in-depth local search and to identify its optimal discriminative features. These optimal local solutions are then concatenated to generate a new swarm leader, which is used to replace the previous leader if it has a better fitness value.

The overall optimisation process of our algorithm iterates until (1) the number of evolution reaches 500; (2) the fitness value does not show obvious improvement during the last 50 generations. The proposed PSO-based feature selection is conducted for each emotion category separately to identify discriminative features for each expression. The

generated optimal feature subset of each expression by our PSO algorithm is shown in Figure 5.4, with a detailed analysis provided in the Evaluation section. Empirical results indicate that our algorithm outperforms other PSO variants and conventional methods regarding the search towards global optimum and discriminative feature selection (see Section 5.2).

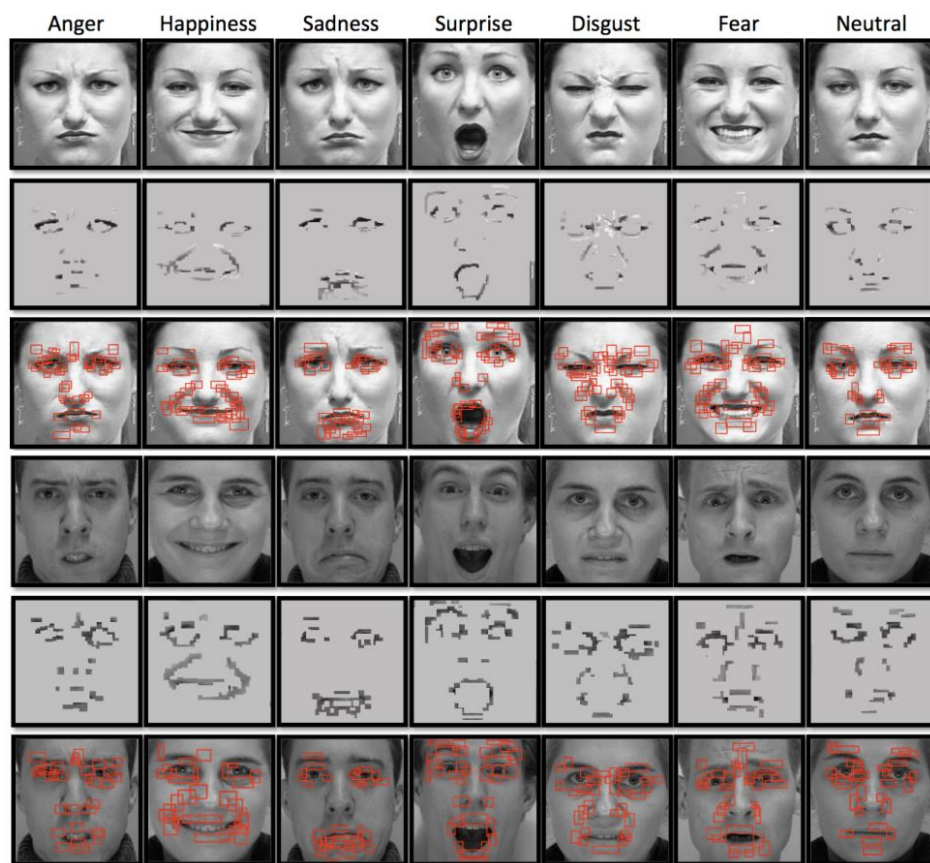


Figure 5.4. The selected optimal features and their distribution for each expression using the proposed mGA-embedded PSO algorithm (Rows 1-3: CK+ images, Rows 4-6: MMI images)

5.1.3 Emotion recognition

In this research, we conduct a study of 7-class facial emotion recognition using the features automatically generated by the mGA-embedded PSO. NN with Backpropagation, a multi-class SVM (Crammer & Singer, 2001), and ensemble classifiers are used for classification. The detailed setting of the classifiers is introduced, as follows. In this research, the trial-and-error method is conducted to identify the optimal NN structure,

whereas a grid-search method is applied to find the optimal parameters of the multi-class SVM with the RBF kernel. After several trials, the NN is equipped with one input layer with 25-40 nodes indicating the optimised features obtained from the proposed PSO algorithm, one hidden layer, and one output layer with seven nodes, respectively, representing seven expressions. For the grid search of optimal settings for the multi-class SVM with the RBF kernel, we use exponentially growing sequences and search the ranges of $[2^{-5}, -2^{15}]$, $[2^{-10}, -2^5]$, and $[2^{-8}, -2^{-1}]$, respectively, for a soft-margin constant, C , a kernel parameter, gamma (γ), and an epsilon (ϵ) in the loss function since the combination of these three parameters plays very important roles in affecting the SVM's performance. We also employ 10-fold cross validation to identify the best combination of these parameters to avoid over-fitting. The identified optimal setting in the training stage is then applied to the subsequent experiments in the test stage.

Besides these single model classifiers, we also employ ensemble classifiers for expression recognition in order to improve accuracy. We use weighted majority voting for the construction of ensembles because of its impressive performance and suitability for undertaking small datasets (<1000) in this research. We construct two ensembles with NN and multi-class SVM as the base model, respectively. Also the NN-based and SVM-based ensembles use three base models respectively. The optimal settings identified earlier for NN and SVM are applied for building each base model.

The ensemble classifiers are constructed using an AdaBoost process so that the performance of the three base models within each ensemble classifier is complementary to each other (Zhang et al., 2015; Neoh et al., 2015). The training process of each ensemble classifier focuses on misclassified instances. As an example, the weights of misclassified instances by the first base model are increased so that they are more likely to be selected for training the second base model. A similar case is also applied to the construction of the third base model, which employs the instances misclassified by the second base model for training. Therefore, each ensemble classifier is constructed with a number of base models that are complementary to each other (Zhang et al., 2015; Neoh et al., 2015). Weighted majority voting is applied to combine the outputs from the three base models to generate the final output for each ensemble. The empirical results indicate that the constructed

ensembles outperform NN/SVM based emotion recognition for both within and across database evaluations.

5.2 Evaluation

In this research, both CK+ and MMI are employed for evaluation. A set of 250 images from CK+ is used for training while 175 images extracted from CK+ and MMI, respectively, are employed for testing.

5.2.1 Comparison of feature extraction algorithms

First of all, a series of experiments is conducted to compare the proposed *hvnLBP* operator with other state-of-the-art texture descriptors including CLBP, DLBP, CS-LBP, LDP and LPQ. The Gabor filter is integrated with each texture descriptor algorithm for feature extraction. Low-level raw features extracted by each descriptor are directly used for emotion classification without any feature optimization. When ensemble classifiers are applied, all algorithms achieve the best performance. Table 5.1 shows the evaluation results of all descriptors integrated with ensembles with each ensemble trained with features extracted by each texture descriptor. Both second and third-order LDPs are implemented. The results of the second-order LDP are presented in Table 5.1, since it achieves the best performance.

Table 5.1 Comparison between the *hvnLBP* operator and other texture descriptors.

| CK+(train)/CK+(test) | CK+(train)/MMI(test) |
|----------------------|----------------------|
|----------------------|----------------------|

| Methods | Ensemble | Ensemble | Ensemble | Ensemble |
|------------------------|----------|----------|----------|----------|
| | (NN) % | (SVM) % | (NN) % | (SVM) % |
| LBP | 75.00 | 77.67 | 62.98 | 64.34 |
| DLBP | 76.12 | 77.95 | 63.54 | 65.23 |
| LPQ | 76.00 | 78.27 | 63.90 | 65.09 |
| CLBP | 78.56 | 79.54 | 64.10 | 66.77 |
| CS-LBP | 80.19 | 81.36 | 65.70 | 68.66 |
| LDP (2nd order) | 85.92 | 86.78 | 71.05 | 74.95 |
| <i>hvnLBP</i> | 89.00 | 90.60 | 73.20 | 77.13 |

The empirical results in Table 5.1 indicate that *hvnLBP* possesses more discriminative capabilities and outperforms all the selected state-of-the-art LBP variants, LPQ, and conventional LBP significantly for both within and cross database evaluations. When the SVM-based ensemble classifier is applied, all algorithms achieve the best accuracy rates, and *hvnLBP* outperforms LBP, DLBP, LPQ, CLBP, CS-LBP, and LDP by 12.93%, 12.65%, 12.33%, 11.06%, 9.24%, and 3.82%, respectively, for within-database evaluation and by 12.79%, 11.9%, 12.04%, 10.36%, 8.47%, and 2.18%, respectively, for cross-database evaluation.

Built upon the LBP methodology, DLBP and CLBP rely on the comparison between the centre point and its neighbours but ignore the differences among neighbourhood pixels themselves. Therefore, they show limitations in identifying different local structures embedded in the neighbouring pixels. CS-LBP employs centre-symmetric pixel pairs for comparison, in order to extract local discriminative information. However, it overlooks other local differences among horizontal and vertical pixels. An example that demonstrates the difference among the proposed *hvnLBP* operator, LBP, and CS-LBP is provided, as follows. Given two patterns (50, 80, 85, 70, 50, 45, 55, 53, centre-60) and (100, 230, 240, 230, 100, 50, 120, 160, centre-200), although the local structures of both patterns are different, LBP generates the same binary code, 01110000, for both patterns. CS-LBP produces 11111000 for both patterns too. However, *hvnLBP* is able to generate two distinctive binary codes for these patterns, i.e. 01110010 for the former and 01110011 for the latter, indicating the two different local structures.

LPQ with decorrelation is implemented in our experiment. LPQ shows great robustness to blurred images by employing local phase information calculated using a short-term Fourier transform for each pixel position. However, it has higher computational complexity, and is expensive for online applications in comparison with *hvnLBP*. In addition, the window size is one of the important parameters in LPQ. A smaller window is able to capture detailed texture information, but other unimportant patterns caused by illumination changes and noise factors are extracted as well. On the contrary, a larger window sometimes is not able to extract sufficient discriminative information, therefore decreasing the performances for sharp images (Ojansivu & Heikkila, 2008).

Among all the comparable descriptors, the second-order LDP achieves the best accuracy rate, which extracts more detailed high-order local pattern information. However, the empirical results indicate that sometimes it also extracts over-detailed patterns which contain more noise in comparison with *hvnLBP*. Moreover, the second-order LDP also generates high dimensional features with a high computational cost, which makes it less suitable for real-time applications. Another limitation of using LDP is the requirement of identifying the optimal order of LDP that is suitable for a specific database although the third-order LDP outperformed all the other order LDPs in Zhang et al. (2010) for face recognition tasks.

In comparison with the above-mentioned comparable methods, the proposed *hvnLBP* operator effectively extracts spatial relationships in a local region by conducting multiple direct horizontal and vertical neighbourhood comparisons with an efficient computational cost. From the empirical study, it shows superior capabilities of preserving distinctiveness and differentiating different local structures embedded in the neighbouring pixels for low contrast images.

5.2.2 Comparison of feature selection algorithms

To evaluate the proposed mGA-embedded PSO algorithm for feature selection, we have implemented state-of-the-art methods for comparison, i.e. ELPSO (Jordehi, 2015), a PSO variant for multimodal function optimization (i.e. MFOPSO in this paper) (Chang, 2015), Binary bare bones PSO (i.e. BBPSO in this paper) (Zhang et al., 2015), Threshold-

based Binary PSO (i.e. ThBPSO) (Krisshna et al., 2014), high exploration PSO (i.e. HEP SO) (Mehmoodabadi et al., 2014), conventional PSO, and classical GA. The features extracted by *hvnLBP* are further processed by each feature optimization algorithm for dimensionality reduction. NN, SVM, and NN-based and SVM-based ensembles are applied to recognize seven emotions using automatically generated features based on each feature optimization technique.

Owing to the stochastic property of the proposed PSO algorithm and other compared methods, we have performed 30 runs for each method integrated with each classifier for within and cross database evaluations, respectively. First of all, we conduct the within-database evaluation by applying 250 and 175 images from CK+ for training and testing, respectively. Table 5.2 shows the average classification performance of 30 runs for all optimisation algorithms in combination with diverse classifiers. The best results are obtained for each feature selection model when the SVM-based ensemble is applied. The proposed mGA-embedded PSO algorithm achieves an average accuracy rate of 100% for seven emotions, and outperforms seven other algorithms by 2.6% (BBPSO), 2.7% (MFOPSO), 4.7% (ELPSO), 5.6% (HEPSO), 7.4% (ThBPSO), 14.7% (PSO), and 20.2% (GA), respectively. Moreover, our algorithm extracts a comparatively smaller set of features (25-40) with fewer computational costs.

Table 5.2 Average classification performance using the selected optimisation algorithms integrated with diverse classifiers over 30 runs respectively for within database evaluation.

| Methods | No. of selected features | NN % (30 runs) | SVM % (30 runs) | Ensemble (NN) % (30 runs) | Ensemble (SVM) % (30 runs) |
|---------|--------------------------------|----------------------|-----------------------|---------------------------------|-------------------------------------|
|---------|--------------------------------|----------------------|-----------------------|---------------------------------|-------------------------------------|

| | | | | | |
|------------------|---------|-------|-------|-------|------|
| GA | 150-220 | 74.06 | 77.73 | 78.03 | 79.8 |
| PSO | 150-200 | 75.46 | 78.10 | 83.5 | 85.3 |
| ThBPSO | 80-120 | 85.86 | 85.16 | 89.3 | 92.6 |
| HEPSO | 70-100 | 86.50 | 87.16 | 93.80 | 94.4 |
| ELPSO | 65-80 | 87.26 | 88.43 | 94.86 | 95.3 |
| MFOPSO | 70-110 | 87.90 | 89.16 | 95.60 | 97.3 |
| BBPSO | 50-100 | 88.55 | 89.88 | 96.33 | 97.4 |
| Prop. PSO | 25-40 | 94.33 | 95.50 | 100 | 100 |

Table 5.3 Average classification performance using the selected optimisation algorithms integrated with diverse classifiers over 30 runs respectively for cross-database evaluation.

| Methods | No. of Selected features | NN % (30 runs) | SVM % (30 runs) | Ensemble (NN) % (30 runs) | Ensemble (SVM) % (30 runs) |
|------------------|---|-------------------------------|--------------------------------|--|---|
| GA | 150-220 | 70.13 | 71.23 | 75.11 | 76.31 |
| PSO | 150-200 | 70.90 | 72.50 | 76.05 | 76.77 |
| ThBPSO | 80-120 | 76.75 | 78.56 | 81.30 | 82.38 |
| HEPSO | 70-100 | 78.55 | 79.76 | 83.50 | 85.17 |
| ELPSO | 65-80 | 78.84 | 80.33 | 86.00 | 87.45 |
| MFOPSO | 70-110 | 80.85 | 81.76 | 87.60 | 88.09 |
| BBPSO | 50-100 | 81.07 | 81.88 | 87.80 | 88.31 |
| Prop. PSO | 25-40 | 88.97 | 90.70 | 92.90 | 94.66 |

We have also conducted the cross-database evaluation with a training set of 250 images from CK+ and a test set of 175 images from MMI. Table 5.3 summarises the average accuracy rates for all the selected models integrated with different classifiers over 30 runs for the cross-database evaluation. The best performances are yielded by the SVM-based ensemble for all feature selection methods. The proposed PSO algorithm extracts the smallest number of features, achieves an average accuracy rate of 94.66% for seven emotions, and outperforms seven other methods by 6.35% (BBPSO), 6.57% (MFOPSO), 7.21% (ELPSO), 9.49% (HEPSO), 12.28% (ThBPSO), 17.89% (PSO), and 18.35% (GA), respectively. In Figure 5.5, the boxplot diagrams clearly demonstrate the distribution of the classification results over 30 runs of all the feature selection methods in combination with the SVM-based ensemble for the cross-database evaluation.

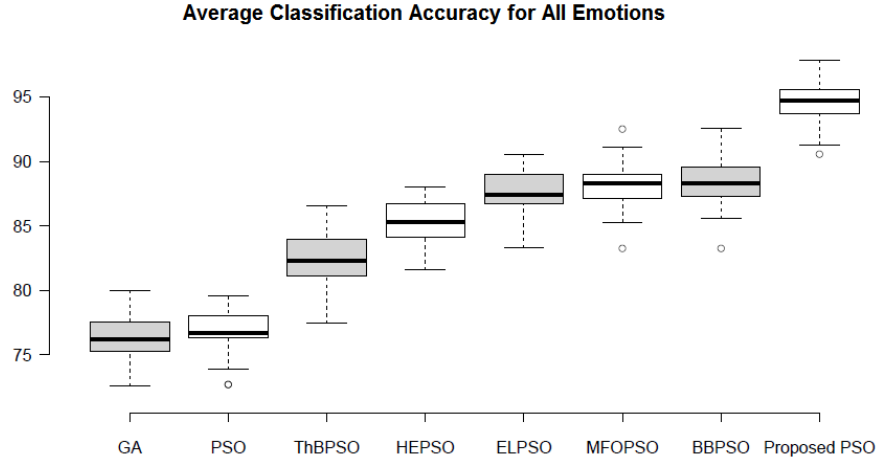


Figure 5.5. Boxplot diagram for the distribution of average recognition results for each optimisation algorithm + SVM-based ensemble over 30 runs for cross-database evaluation

As can be seen in Figure 5.5, the results of all 30 runs of the proposed PSO algorithm outperform those of all other state-of-the-art PSO variants, conventional PSO, and classical GA by a significant margin. E.g., all the results of 30 runs of our algorithm except for one outlier (with the lower whisker at 91.29%) are higher than the maximum results of all the following methods, i.e. 91.14% for MFOPSO, 90.57% for ELPSO, 88% for HEPsO, 86.57% for ThBPSO, 79.57% for PSO, and 80% for classic GA. Furthermore, at least 75% of the results of our algorithm (with the first quartile of 93.71%) are higher than the maximum result, i.e. 92.57% from BBPSO. Among all the selected state-of-the-art PSO variants, BBPSO, MFOPSO and ELPSO achieve comparatively better performances than HEPsO and ThBPSO, i.e. with at least 25% of the results of these three PSO variants higher than the maximum result (88%) of HEPsO and at least 75% of the results of these three PSO variants higher than the maximum result (86.57%) of ThBPSO. In comparison with these three best PSO variants, i.e. BBPSO, MFOPSO and ELPSO, the median value of our algorithm (94.71%) is higher than the median scores of BBPSO (88.29%), MFOPSO (88.29%), and ELPSO (87.43%) by 6.42%, 6.42%, and 7.28%, respectively. Besides outperforming these three best PSO variants, all the results of our algorithm are within a smaller variation range of [91.29%, 97.86%], as compared with those from BBPSO having a larger variation of [85.57%, 92.57%]. Moreover, the lowest result of our PSO algorithm (i.e. the lower whisker at 91.29%) outperforms the maximum results of HEPsO (88%),

ThBPSO (86.57%), classical GA (80%) and PSO (79.57%) by 3.29%, 4.72%, 11.29%, and 11.72%, respectively.

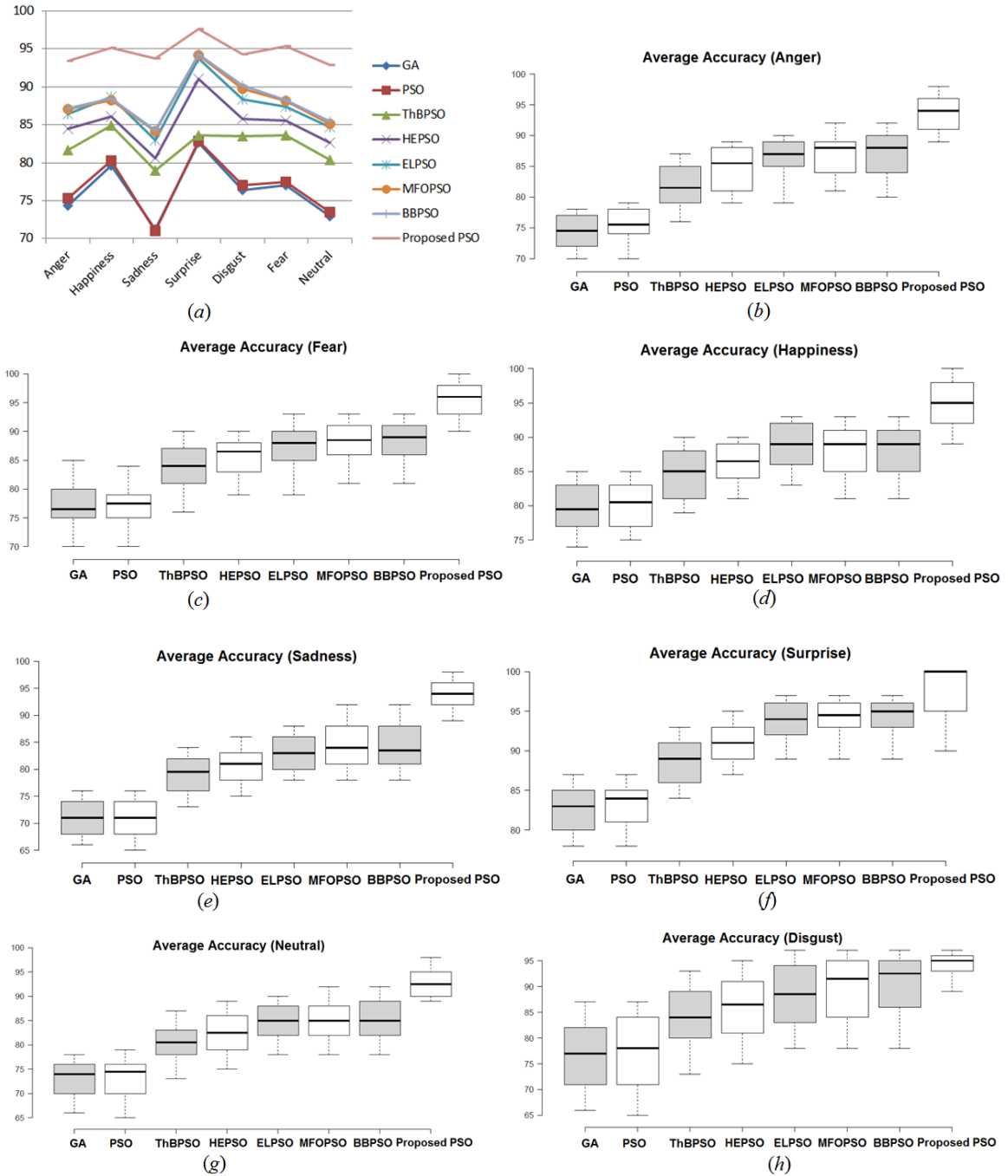


Figure 5.6. (a) Overall comparison of our system with other methods and (b) – (h) Boxplot diagrams for the distribution of classification results for each emotion category for each optimisation algorithm + SVM-based ensemble over 30 runs for cross-database evaluation

Furthermore, the average classification results of each expression over the 30 runs for each optimisation method with the SVM-based ensemble classifier for the cross-database evaluation are depicted in Figure 5.6 (a). Figure 5.6 (b)-(h) indicate the detailed boxplot diagrams for the distribution of the detailed classification results over 30 runs for each emotion category. As indicated in Figures 5.6 (a)-(h), the proposed PSO algorithm achieves superior performance and outperforms all the other compared methods for each emotion significantly. With respect to the fear and sadness emotion categories, 75% of the classification results of our model are higher than the maximum results of all seven methods, whereas at least 50% of the results of our algorithm are also higher than the maximum results of all other methods for the anger, happiness, surprise, and neutral emotion classes. Meanwhile, for the disgust emotion, the results of our algorithm over 30 runs indicate the overall smallest variation of [89%, 97%], as compared with other larger variations of the other results, e.g. [78%, 97%] for BBPSO, MFOPSO, and ELPSO, respectively. The proposed diversity maintenance strategies of our PSO algorithm contribute to its superior performance over other state-of-the-art and conventional methods.

An analysis of the algorithmic contribution of the proposed approach is as follows. We compare our PSO algorithm with the three advanced PSO variants, i.e. BBPSO, MFOPSO, and ELPSO.

BBPSO (Zhang et al., 2015) employs a reinforced memory strategy for updating *pbest* for each particle and a uniform combination technique to replace sub-dimensions of each particle using a random number with the corresponding elements of a randomly selected *pbest_k* from a set of stored *pbests* to avoid stagnation. It increases the execution of a uniform combination with respect to increased stagnant iterations. However, since the uniform combination operation is only applied to the sub-elements of swarm particles and simulates the effects of crossover and mutation operations of the GA, the generated offspring could be significantly similar (i.e. with a high correlation) to the parent particles. Therefore, their search strategy focuses more on local exploitation. In contrast, our PSO variant applies follower particles which have the highest or lowest correlation with the leader to diversify the search and increase both local and global search capabilities, in an

attempt to avoid stagnation. Therefore, it shows a superior performance than that of BBPSO.

MFOPSO (Chang, 2015) divides the original swarm into several sub-swarms to increase search diversity. It is capable of dealing with multimodal function optimisation. However, when the search fails to generate fitter leaders in the sub-swarms, MFOPSO does not include any diversity maintenance or jump out mutation strategy to diversify the search in the sub-swarms, in order to avoid premature convergence.

Table 5.4 Comparison with related research on the CK+ database.

| | Method | Classes | Dynamic | Measure | Recog. Rate (%) |
|--|---|---------|---------|---|--------------------|
| Shojaeilan gari et al. (2014) | Histogram of Dominant Phase Congruency (HDPC) + SVM | 6 | Y | Leave-one- subject-out | 95.44 |
| Liu et al. (2015) | Action Units inspired Deep Networks | 7 | N | 10-fold | 93.70 |
| Xiao et al. (2011) | Multiple manifolds | 6 | N | 25% for testing | 96.57 |
| Neoh et al. (2015) | Overlap LGBP + direct similarity + weighted majority vote | 7 | N | 42.8% for testing | 96.80 |
| Neoh et al. (2015) | Overlap LGBP + Pareto + weighted majority vote | 7 | N | 42.8% for testing | 97.40 |
| This research | <i>hvnLBP</i> + proposed mGA- embedded PSO + ensemble (SVM) | 7 | N | Average of 30 runs with 46.6% for testing for each run | 100 |

Table 5.5 Comparison with related work development on the MMI database.

| | Method | Classes | Dynamic | Measure | Recog. Rate (%) |
|--------------------------------|--|---------|---------|--|--------------------------|
| Fang et al. (2014) | Group-wise registration + parametric space + fuzzy-rough based nearest neighbour | 6 | Y | 10-fold | 75.96(trained with MMI) |
| Fan and Tjahjadi (2015) | Histogram of gradients + optical flow + SVM | 7 | Y | 10-fold | 58.7 (trained with CK+) |
| Liu et al. (2015) | Action Units inspired Deep Networks | 7 | N | 10-fold | 72.2 (trained with CK+) |
| This research | <i>hvnLBP</i> + proposed mGA-embedded PSO + ensemble (SVM) | 7 | N | Average of 30 runs with 70% for testing for each run | 94.66 (trained with CK+) |

The same explanation applies to ELPSO (Jordehi, 2015). It employs Gaussian, Cauchy, opposition-based, and DE-based mutation strategies to increase the exploration capability of the swarm leader. However, ELPSO only attempts to improve the leader when stagnation occurs, and no improvement strategy is applied to the follower particles to retain population diversity. In comparison with MFOPSO and ELPSO, our PSO algorithm utilises the diversity maintenance mechanism of mGA and keeps a non-replaceable memory to maintain swarm diversity. It not only applies Gaussian mutation to the swarm leader to enable long jumps in the primary swarm but also employs particles with the highest or lowest correlation with the swarm leader from the non-replaceable memory to retain population diversity and increase local exploitation and global exploration. Most importantly, these local and global search strategies of the secondary swarm work collaboratively to lead the search towards the global optimum. Therefore, it outperforms MFOPSO and ELPSO significantly regarding achieving the global optimum and enabling discriminative feature selection.

5.2.3 Comparison of emotion recognition systems

A comparison between our proposed PSO algorithm and other recent state-of-the-art facial expression recognition methods has been conducted. Tables 5.4 and 5.5 show the comparison among different methods using the CK+ and MMI databases, respectively. As shown in Table 5.4, for the evaluation using CK+, Neoh, et al. (2015), which proposed both direct similarity and Pareto-based optimisation for facial feature selection, achieves the best performance. The Pareto-based feature selection emphasises both intra-class and inter-class variations and achieves the highest accuracy rate. However, although related strategies are adopted in their fitness functions to prevent information loss, inspection of their results indicate that the algorithms produce a comparatively small subset of 13-39 features and, sometimes, could overlook certain important features pertaining to certain emotion categories (e.g. widened eyes for surprise, mouth stretch for fear, etc) in comparison with our proposed algorithm. As illustrated in Figure 5.4, the feature sub-regions extracted by our PSO algorithm indicate the most significant texture distortions around the eyes, eyebrows, and the mouth associated with each distinct expression. The key facial muscular actions defined in FACS (Ekman et al., 2002) associated with each expression can be clearly seen in the optimised features revealed by our algorithm. E.g., for anger, significant features indicating brow lower, eyelid and lip tightener are produced by our PSO algorithm, whereas the sub-regions indicating the significance of lip corner puller and cheek raiser are revealed for the happy expression. Feature distribution pertaining to sadness clearly indicates the implication of the inner brow raiser and lip corner depressor whereas eyebrow raiser, widened eyes and mouth open are demonstrated in the selected sub-regions for surprise, etc. Overall, the features identified by our PSO algorithm represent the characteristics of each emotion significantly and map closely to the AUs given in FACS.

We also conduct the cross-database evaluation to further assess the scalability of the proposed PSO algorithm using the MMI database. Table 5.5 shows a comparison with other related methods. Fang et al. (2014) employed MMI for both training and testing, whereas other methods including our work used CK+ for training and MMI for testing. Results indicate our algorithm shows great scalability and extracts the most discriminative features

of each expression for the cross-domain evaluation. It outperforms all related methods by a significant margin of approximately 20-37%.

5.3 Summary

In this Chapter, we have proposed a facial expression recognition system with *hvnLBP* based feature extraction, mGA-embedded PSO-based feature optimization and diverse classifier based expression recognition. The proposed *hvnLBP* operator performs horizontal and vertical neighbourhood pixel comparison to retrieve the initial discriminative facial features. It outperforms state-of-the-art LBP variants, LPQ, and conventional LBP significantly for texture classification. Moreover, a new PSO algorithm, i.e. mGA-embedded PSO, has been proposed to mitigate the premature convergence problem of conventional PSO in terms of feature optimization. The mGA-embedded PSO algorithm incorporates personal average experience and Gaussian mutation for velocity updating as well as employs the diversity maintenance strategy of mGA by keeping the original swarm in a non-replaceable memory, which remains intact during the lifecycle of the algorithm to increase swarm diversity. Furthermore, it also maintains a secondary swarm with a small population size of five to host the swarm leader and four follower particles with the highest/lowest correlation with the leader from the non-replaceable memory to increase local and global search capabilities. The algorithm subsequently separates facial features into specific areas for in-depth local sub-dimension based search. Overall, the local exploitation and global exploration search mechanisms of the algorithm work cooperatively to guide the search towards the global optimal solutions. The empirical results indicate that our PSO algorithm outperforms other state-of-the-art PSO variants and conventional PSO and GA for optimal feature selection significantly. The empirical results also indicate that our proposed PSO algorithm outperforms other related facial expression recognition methods reported in the literature by a significant margin.

Chapter 6 Conclusion and Future Work

In this research, we have focused on automatic expression recognition system with various facial feature extraction and feature optimisation methods. In this chapter, we have summarised the contributions of this research and also stated possible future improvements.

6.1 Summary of Contributions

In this research, a number of contributions have been stated to solve problems related to automatic facial expression recognition. First of all, we propose a novel AAM with BRISK model, which is used to extract facial features for the estimation of the AU intensity and automatic expression recognition. We also have proposed an unsupervised facial point detector with FCM based occlusion detection for AU intensity estimation and facial expression recognition. The evaluation results show that the proposed unsupervised facial point detector outperforms other related state-of-the-art methods. We have also proposed a texture-based facial expression recognition system, which employs a modified LBP feature extraction and mGA-embedded PSO feature optimisation. The empirical findings show that the evolutionary optimisation based feature selection techniques benefit the performance of the automatic facial expression recognition system significantly. The detailed discussions on research contributions are presented below.

6.1.1 A supervised feature extraction model using AAM in combination with BRISK

An intelligent facial action and emotion recognition system is proposed. This system estimates the intensity of the selected 18 AUs and recognises seven basic emotions. A novel feature extraction model using AAM combined with BRISK is employed in order to extract the facial shape-based and texture-based features from the input image. In comparison to the original AAM algorithm, the proposed model is more robust to rotation variations and scaling differences. In order to further improve the representation of texture features, the

LGBP operator is applied to extract appearance features obtained from AAM in combination with BRISK.

Furthermore, shape and texture based NN AU analysers are used respectively to estimate the intensity of the selected 18 AUs. The derived AUs are used as input to an NN-based emotion recognizer. The evaluation results prove that the proposed system can achieve high accuracy for AU and emotion recognition.

6.1.2 Unsupervised facial point detector for facial emotion recognition

In order to detect emotions from images with pose variations, illumination changes, occlusions and background noise, we have proposed a computationally cost-effective superior unsupervised facial point detector model. This model employs various algorithms such as 2D Gabor filter, a novel feature descriptor, BRISK, the ICP algorithm and FCM to retrieve facial landmark points. In order to deal with images with occluded facial regions, the proposed facial point detector applies ICP to first recover neutral landmark points for the occluded region. Then FCM is applied to further improve the geometry of the occluded region by taking prior knowledge of the attributes of the non-occluded facial regions into account. We then select the top five image outputs with the highest correlations to the test image in the cluster and average them to re-construct the best fitting geometry for the occluded facial element. The reconstructed set of landmarks with shape information embedded for the occluded facial region is then used to adjust the neutral landmarks generated by ICP.

This background robust facial feature point detector can detect 54 facial points for images or real subjects from challenging real-time human-computer interaction with great efficiency. The SVR and NN classifiers are employed in order to estimate the intensity of the selected 18 AUs. FCM is employed respectively in order to recognise the seven basic emotions. The proposed system also shows great potential to deal with compound and newly arrived novel emotion class detection. The evaluation results show that the proposed system outperforms other related state-of-the-art algorithms.

6.1.3 The texture based facial emotion recognition using *hvnLBP* feature extraction and mGA-embedded PSO feature selection

In order to select more discriminative features and reduce the feature dimensions, we have proposed *hvnLBP* for feature extraction and mGA-embedded PSO feature optimization. The *hvnLBP* algorithm conducts horizontal and vertical neighborhood pixel comparison in order to retrieve the missing contrast information embedded in the neighborhood to generate the initial discriminative facial representation. Compared to original LBP, *hvnLBP* retrieves better contrast information with more discriminative features. A novel mGA-embedded PSO algorithm is proposed for feature selection in order to deal with premature convergence and local optimum problems of conventional PSO. In comparison with other state-of-the-art optimization algorithms, the proposed algorithm proves to have high exploration and exploitation capability.

Multiple classifiers such as single NN, single SVM, NN based ensemble and SVM based ensemble are employed for recognising the seven basic facial expressions. The empirical findings show that SVM-based ensemble can achieve the highest detection accuracy. The proposed system is evaluated using within, and cross-database images from CK+ and MMI databases and the empirical results indicate that the proposed system outperforms other state-of-the-art PSO variants, classical GA, and other related facial expression recognition research.

6.2 Limitations of this Approach

In this section, we discuss some limitations of the presented system. The unsupervised facial point detector model purely focuses on extracting facial shape information, which sometimes may not be sufficient enough for accurately classifying facial expressions. Also, the current unsupervised facial point detector model can only deal with rotations up to 60° and in future, it will be extended to deal with rotations up to 90° . The current mGA embedded PSO based facial emotion recognition system can only deal with non-occluded facial images, which also motivates directions for future research.

6.3 Future Work

In this section, we have discussed some potential future advancements of the proposed research. For instance, we aim to extend our unsupervised facial point detector model to deal with emotion recognition under extreme conditions such as crying and yawning. The performance of the proposed unsupervised facial point detector will be validated on more challenging real-life images extracted from real-world situations.

Furthermore, other evolutionary optimisation algorithms will also be investigated (e.g. firefly algorithm and Cuckoo search algorithm) for feature optimisation. Multi-objective evolutionary algorithms will also be explored to further improve the current system to deal with real-world challenging optimisation problem containing multiple criteria. Different ensemble construction models using base models trained on features provided by LBP variants and other feature extraction techniques can be explored to further improve the emotion recognition performance. In future, we aim to deploy the proposed system to other platforms such as smart-phone devices, surveillance cameras and humanoid robot platforms, to promote brain-computer interfaces and enhance user experience. Some possible lines of long-term future directions are also presented below.

6.3.1 Combining bodily gestures with facial expressions to enhance emotion recognition performance

Empirical findings show that in some cases facial expressions may not be informative enough for reliable emotion interpretation. A more robust emotion recognition system taking both facial and bodily expressions into account should be considered to differentiate different emotion categories more reliably in diverse spontaneous social interactions. Literature shows that very limited research has been conducted on emotion recognition using combined bodily gestures and facial expression features (Zhang et al., 2015). In future work, this line of research will be explored.

6.3.2 Exploring micro-emotions

Another interesting area for future work is to recognise micro expressions. Sometimes the subtle facial reactions exposed unwillingly also hold significant information

representing a specific emotion. For example, when someone makes attempts to hide his/her facial expression, it is highly possible to expose a micro expression. Therefore, an application capable of recognising such micro expressions will surpass the current system and probably capability of most humans.

6.3.3 Exploring various classifiers for emotion recognition

Furthermore, the literature shows that evolutionary optimisation algorithms can significantly improve the performance of the single classifiers such as ANN and SVM by identifying optimal parameter settings (Pendharkar, 2012; Sajan et al., 2015). Research also indicates that evolutionary optimisation based ensemble construction can significantly improve the classification performance of the ensemble classifiers (Zavaschi et al., 2013). In order to further improve emotion recognition accuracy, we will explore both such metaheuristic search algorithm guided parameter tuning for single classifiers and ensemble construction models in future directions. Another interesting and evolving research area worthy of exploring for facial expression recognition is deep learning techniques. Many deep learning architectures have been proposed for diverse machine vision problems which gained outstanding performance for vision-based tasks in the wild (Schmidhuber, 2015). Because of their impressive performance, deep learning based facial expression recognition will also be explored especially for dealing with spontaneous emotion recognition in the wild.

6.3.4 Various applications for the proposed models

Finally, the proposed system can be integrated with an intelligent conversational agent and/or robotics systems in order to achieve human-like interaction to enhance user experience. The proposed research can also be integrated with personalised healthcare, intelligent tutoring and entertainment systems to benefit the user experience. Another application of the proposed research is to assist autistic children who are struggling to interpret expressions to cope better during social interaction. Moreover, the proposed feature extraction and feature selection techniques can also be easily extended to benefit other application domains such as bioinformatics. For example, the proposed algorithms

can be tuned and applied to cell and brain MRI image segmentation, skin cancer and blood cancer classification.

References List

- Abdullah A., Deris S., Mohamad M., Hashim S., (2012). A new hybrid firefly algorithm for complex and nonlinear problem, *Distributed Computing and Artificial Intelligence*, pp 673–680.
- Abdullah A., Deris S., Anwar S., Arjunan SNV. (2013). An evolutionary firefly algorithm for the estimation of nonlinear biological model parameters, *PLoS One.*; 8(3):e56310.
- Adya, Monica, and Fred Collopy. (1998). How Effective Are Neural Networks at Forecasting and Prediction? A Review and Evaluation. *Journal of Forecasting* 17(5–6): 481–95.
- Ajit Krishna N. L., Kadedotad Deepak V., Manikantan K., Ramachandran S. (2014). Face recognition using transform domain feature extraction and PSO-based feature selection, *Appl. Soft Comput.* 22, pp 141–161.
- Ali S.F., and Ramaswamy A. (2009). Optimal fuzzy logic control for MDOF structural systems using evolutionary algorithms, *Engineering Applications of Artificial Intelligence.* 22: pp. 407–419.
- Alviar J.B., Pena J., Hincapie R., (2007). Subpopulation best rotation: a modification on PSO. *Revista Facultad de Ingenieria* No 40, pp 118–122.
- Alweshah M., Abdullah S., (2015). Hybridizing firefly algorithms with probabilistic neural network for solving classification problem, *Applied Soft Computing* 35 513–524.
- Andrews, Robert, Joachim Diederich, and Alan B. Tickle. 1995. Survey and Critique of Techniques for Extracting Rules from Trained Artificial Neural Networks. *Knowledge-Based Systems* 8(6): 373–89.
- Asthana A., Zafeiriou S., Cheng S., Pantic M. (2013). Robust discriminative response map fitting with constrained local models, in: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. USA, pp. 3444–3451.
- Azazi A., Luthi S.L., Venkat I., (2014). Identifying Universal facial emotion markers for automatic 3D facial expressions recognition. In: *International Conference on Computer and Information Sciences (ICCOINS)*, pp 1–6.
- Bartlett M. S., Littlewort G., Fasel I., Movellan J.R. (2003). Real time face detection and facial expression recognition: development and applications to human computer interaction, in: *IEEE Conference on Computer Vision and Pattern Recognition*, p. 53.
- Bay H., Ess A., Tuytelaars T., Gool L.V. (2008). Speeded up robust features (surf), *J. Comput. Vis.*

- Image Understand. 110 (3) 346–359.
- Belhumeur P., Jacobs D., Kriegman D., Kumar N. (2011). Localizing parts of faces using a consensus of exemplars, in: CVPR, IEEE, 2011, p.
- Belli, M.R, M Conti, P Crippa, and C Turchetti. (1999). Artificial Neural Networks as Approximators of Stochastic Processes. *Neural Networks* 12(4): 647–58.
- Besl P.J., McKay N.D., (1992). A method for registration of 3-d shapes, *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (2) 239–256.
- Burges, Chris J.C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery* 2.
- Cai D., He X., Han J., Zhang H.-J. (2006). Orthogonal laplacian faces for face recognition, *IEEE Trans. Image Process.* 15 3608–3614.
- Callen, Jeffrey L, Clarence C.Y Kwan, Patrick C.Y Yip, and Yufei Yuan. (1996). Neural Network Forecasting of Quarterly Accounting Earnings. *International Journal of Forecasting* 12(4): 475–82.
- Calonder M., Lepetit V., Strecha C., Fua P. (2010). Brief: Binary robust independent elementary features, in: *Proceedings of the 11th European Conference on Computer Vision (ECCV): Part IV*, Springer, pp. 778–792.
- Campos M., Krohling R.A. and Enriquez I. (2014). Bare Bones Particle Swarm Optimization with Scale Matrix Adaptation. *Cybernetics, IEEE Transactions on.* 44(9). pp. 1567–1578.
- Castrilln M., Dniz O., Guerra C., Hernndez M. (2007). Encara2: Real-time detection of multiple faces at different resolutions in video streams, *J. Vis. Commun. Image Represent.* 18 (2) 130–140.
- Castrilln-santana M., Dniz-surez O., Antn-canals L., Lorenzo-navarro J. (2008). Face and facial feature detection evaluation, in: *Proceedings of Third International Conference on Computer Vision Theory and Applications (VISAPP'08), INSTICC - Institute for Systems and Technologies of Information, Control and Communication*, pp. 167–172.
- Castro, J L, C J Mantas, and J M Benitez. (2002). Interpretation of Artificial Neural Networks by Means of Fuzzy Rules. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council* 13(1): 101–16.
- Castro, J.L., C.J. Mantas, and J.M. Benítez. (2000). 13 Neural Networks Neural Networks with a Continuous Squashing Function in the Output Are Universal Approximators.

- Chang C.-C., Lin C.-J. (2011). Libsvm: A library for support vector machines, *ACM Trans. Intell. Syst. Technol.* 2 (3) 1–27. Article No. 27
- Chang W. (2015). A modified particle swarm optimization with multiple subpopulations for multimodal function optimization problems. *Appl Soft Comput.* 33, pp. 170–182.
- Chavan U. B., Kulkarni D. B. (2013). Facial expression recognition-review, *Int. J. Latest Trends Eng. Technol.* 3 (1) 237–243.
- Chen H.Y., Huang C.L., Fu C.M. (2008). Hybrid-boost learning for multi-pose face detection and facial expression recognition, *Pattern Recogn.* 41 (3) 1173– 1185.
- Chuang L.Y., Yang C.H., and Li J.C. (2011). Chaotic maps based on binary particle swarm optimization for feature selection, *Appl Soft Comput.* 11: pp. 239–248.
- Church, Keith B., and Stephen P. Curram. 1996. Forecasting Consumers' Expenditure: A Comparison between Econometric and Neural Network Models. *International Journal of Forecasting* 12(2): 255–67.
- Coelho L. S., de Andrade Bernert D. L., Mariani V. C. (2011). A chaotic firefly algorithm applied to reliability-redundancy optimization, *IEEE Congress on Evolutionary Computation (CEC'11)*, 2011:517-521.
- Coello C.A.C., and Pulido G.T. (2001). A Micro-Genetic Algorithm for Multiobjective Optimization, *Evolutionary Multi-Criterion Optimization*, Vol. 1993, *Lecture Notes in Computer Science.*, pp. 126-140.
- Coello C.A.C., and Pulido G.T. (2005). Multiobjective structural optimization using a microgenetic algorithm. *Structural and Multidisciplinary Optimization*, 30(5)., pp. 388-403.
- Cootes T. F., Edwards G. J., Taylor C. J. (2001). Active Appearance Model, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.23, No. 6, 681-685.
- Cootes T.F., Edwards G.J., Taylor C.J. (1998) Active Appearance Models, in: H. Burkhardt, B. Neumann (Eds.), *Proceedings of the European Conference on Computer Vision*, Vol. 2, Springer, pp. 484–498.
- Crammer K., and Singer Y., (2001) On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines, *Journal of Machine Learning Research*. 2: pp. 265-292.
- Cristinacce D., Cootes T. (2006). Feature detection and tracking with constrained local models, in: *Proceedings of British Machine Vision Conference*, Vol. 3, BMVA Press, pp. 929–938.

- Daugman J., Complete discrete 2-d gabor transforms by neural networks for image analysis and compression, *IEEE Trans Acoust. Speech Signal Process.* 36 (7) 1169–1179.
- DeGroot M.H. (1980). *Probability and Statistics*, second ed., Addison-Wesley.
- Demartines P., Hérault J. (1997). Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets, *IEEE Trans. Neural Netw.* 8 148–154.
- DeMenthon D., Davis L. S. (1995). Model-Based Object Pose in 25 Lines of Code, *International Journal of Computer Vision*, 15, 123-141.
- Diao R., Chao F., Peng T., Snooke N. and Shen Q. (2014). Feature Selection Inspired Classifier Ensemble Reduction. *Cybernetics, IEEE Transactions on*, 44 (8) , pp. 1259–268.
- Donato G., Bartlett M.S., Hager J.C., Ekman P., Sejnowski T.J. (1999). Classifying facial actions, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (10) 974–989.
- Dorn M., Buriol L. S., Lamb L. C. (2011). A Hybrid Genetic Algorithm for the 3D Protein Structure Prediction Problem using a Path-Relinking Strategy, in: *IEEE Congress on Evolution Computation*.
- Ekici, Sami. 2012. Support Vector Machines for Classification and Locating Faults on Transmission Lines. *Applied Soft Computing Journal* 12(6): 1650–58.
- Ekman P., Friesen W.V. (1976). *Pictures of Facial Affect*, Consulting Psychologists Press, Palo Alto, CA.
- Ekman P., Friesen W.V., Hager J.C. (2002). *Facial Action Coding System, The Manual*, Research Nexus Division of Network Information Research Corporation, USA.
- Ekman P., Rosenberg E.L. (2005). *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*, 2nd ed., Oxford University Press, New York.
- Eleftheriadis S., Rudovic O., and Pantic M. (2015). Discriminative Shared Gaussian Processes for Multiview and View-Invariant Facial Expression Recognition, *IEEE Transactions on Image Processing*, 24(1), pp. 189 – 204.
- Esmin A.A.A., Lambert-Torres G., (2012) Application of particle swarm optimization to optimal power systems. *Int J Innov Comput Inf Control (IJICIC)* 8(3 (A)):1705–1716.

- Fan X., and Tjahjadi T. (2015). A spatial-temporal framework based on histogram of gradients and optical flow for facial expression recognition in video sequences, *Pattern Recogn.* 48: pp. 3407-3416.
- Fang H., Parthalin N.M., Aubrey A.J., Tam G.K.L., Borgo R., Rosin P.L., Grant P.W., Marshall D., Chen M. (2014). Facial expression recognition in dynamic sequences: An integrated approach, *Pattern Recogn.* 47 (3) 1271–1281.
- Faraway, Julian, and Chris Chatfield. 2008. Time Series Forecasting with Neural Networks: A Comparative Study Using the Air Line Data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 47(2): 231–50.
- Fister I., Fister Jr I., Yang X., Brest J., (2013). A comprehensive review of firefly algorithms, *Swarm and Evolutionary Computation* 13 34-46.
- Fister Jr I., Fister I., Brest J., Yang X. S. (24–25 May 2012) Memetic firefly algorithm for combinatorial optimization, In: Filipič B, Šilc J, eds. *Bioinspired optimization methods and their applications (BIOMA2012)*, Bohinj, Slovenia. 2012:75-86.
- Fletcher, Desmond, and Ernie Goss. 1993. Forecasting with Neural Networks. *Information & Management* 24(3): 159–67.
- Funahashi, Ken-Ichi. 1989. On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks* 2(3): 183–92.
- Gao X., Su Y., Li X., Tao D. (2010). A review of active appearance models, *IEEE Trans. Syst. Man Cybernet. C: Appl. Rev.* 40 (2) 145–158.
- Garg A., Bajaj R., (2015). Facial emotion recognition and classification using hybridization method. In: *International Journal of Engineering Research and General Science*, Vol. 3.
- Gish, H. 1990. A Probabilistic Approach to the Understanding and Training of Neural Network Classifiers. In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, 1361–64.
- Goldberg D.E. (1989). Sizing Populations for Serial and Parallel Genetic Algorithms, In *Proceedings of the 3rd International Conference on Genetic Algorithms.*, pp. 70-79.
- Gosavi A. P., Khot S. R. (2013). Facial expression recognition using principal component analysis, *Int. J. Soft Comput. Eng.* 3 (4) 258–262.
- Gross R., Matthews I., Baker S. (2004). Constructing and fitting active appearance models with occlusion, in: *Proceedings of the IEEE Conference on Computer Vision Pattern*

Recognition Workshops, Vol. 5, IEEE, pp. 72.

- Gu W., Xiang C., Venkatesh Y.V., Huang D., Lin H. (2012). Facial expression recognition using radial encoding of local Gabor features and classifier synthesis, *Pattern Recogn.* 45 80–91.
- Guo Z., Zhang L., and Zhang D. (2010). A completed modelling of local binary pattern operator for texture classification. *IEEE Transactions on Image Processing*, 19(6), pp. 1657-1663.
- Han J., Kamber M., Pei J. (2011). *Data Mining: Concepts and Techniques*, third ed., Morgan Kaufmann.
- Happy S.L., and Routray A. (2015). Automatic facial expression recognition using features of salient facial patches, *IEEE Transactions on Affective Computing*, 6(1), pp. 1 – 12.
- He X., Niyogi P. (2003). Locality preserving projections, *Advances in Neural Information Processing Systems (NIPS)*.
- Heikkilä M., Pietikäinen M., and Schmid C. (2009). Description of interest regions with local binary patterns, *Pattern Recog.*, 42(3), pp. 425–436.
- Henriksen J.J. (2007). 3D surface tracking and approximation using Gabor filters, South Denmark University, (Master dissertation).
- Hippert, H.S., C.E. Pedreira, and R.C. Souza. 2001. Neural Networks for Short-Term Load Forecasting: A Review and Evaluation'. *IEEE Transactions on Power Systems* 16(1): 44–55.
- Holland J. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI, USA: University of Michigan Press .
- Hosseini-Nezhad, Seyed M. et al. 1995. A Neural Network Approach for the Determination of Interhospital Transport Mode. *Computers and Biomedical Research* 28(4): 319–34.
- Hsu C., Chang C., Lin C. (2010). *A Practical Guide to Support Vector Classification*, Department of Computer Science National, Taiwan University.
- Huang D., Shan C., Ardabilian M., Wang Y., Chen L. (2011) . Local binary patterns and its application to facial image analysis: a survey, *IEEE Trans. Syst., Man, Cybern.* – 233. Part C: Appl. Rev. 41 (6) 765–781.
- Huang G.B., Ramesh M., Berg T. (2007) E. Learned-miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, Tech. Rep. 07-49, University

of Massachusetts, Amherst, MA, USA.

- Jain S., Hu C., Aggarwal J.K. (2011). Facial expression recognition with temporal modeling of shapes, in: IEEE International Conference on Computer Vision Workshops (ICCV Workshops), Barcelona, pp. 1642–1649.
- Jaiswal S., Almaev T., Valstar M. F. (2013). Guided unsupervised learning of mode specific models for facial point detection in the wild, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops, IEEE, pp. 370– 377.
- Jordehi A.R. (2015). Enhanced leader PSO (ELPSO): A new PSO variant for solving global optimisation problems. *Appl Soft Comput.* 26, pp. 401–417.
- Juang C.-F. (2004). A hybrid of genetic algorithm and particle swarm optimization for recurrent network design, *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* 34 (2) 997–1006.
- Kaltwang S., Rudovic O., Pantic M. (2012). Lecture Notes in Computer Science, Advances in Visual Computing, Vol. 7432, Springer, Heidelberg, pp. 368–377.
- Kanade T., Cohn J.F., Tian Y. (2000). Comprehensive database for facial expression analysis, in: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), Grenoble, France., p. 4653.
- Karaboga D. (2005). An idea based on honey bee swarm for numerical optimization, Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.
- Kashan M. H, Nahavandi N, Kashan A. H. (2011). Disabc: A new artificial bee colony algorithm for binary optimization. *Appl Soft Comput* doi:10.1016/j.asoc.2011.08.038
- Kasiski A., Florek A., Schmidt A. (2008). The put face database, *Image Process. Commun.* 13 (3) 59–64.
- Kaur M., Vashisht R., Neeru N. (2010). Recognition of facial expressions with principal component analysis and singular value decomposition, *Int. J. Soft Comput. Eng.* 9 (12) 36–40.
- Kennedy J., and Eberhart R. (1995) Particle swarm optimization. In *Proc. IEEE Int. Conf. Neural Network*, Volume 4, pp. 1942–1948.
- Kennedy J., Eberhart R.C., (1995). Particle swarm optimization. In: *IEEE internal conference on neural networks*. Perth, Australia, vol 4, pp 942–1948.

- Kennedy J., Eberhart R.C., (1997). A discrete binary version of the particle swarm algorithm. In: IEEE Conference on systems, man, and cyber, vol 5. pp 4104–4108.
- Kirkpatrick S., Gelatt CD., Vecchi MO. (1983). Optimization by simulated annealing, *Science* ; 220 (4598): 671-680.
- Kohonen T. (Ed.) (1997). *Self-Organizing Maps*, 2nd edition, Springer, Berlin, Germany.
- Krishnakumar K. (1990) Microgenetic algorithms for stationary and nonstationary function optimization, *Proc. SPIE.*, 1196:, pp. 289-296.
- Krishna N.L.A., Deepak V. K., Manikantan K., and Ramachandran S. (2014). Face recognition using transform domain feature extraction and PSO-based feature selection, *Appl Soft Comput.* 22: pp.141–161.
- Lajevardi S.M., Hussain Z.M. (2012). Automatic facial expression recognition: feature extraction and selection, *Signal Image Video Process.* 6 159–169.
- Le V., Brandt J., Lin Z., Bourdev L., Huang T.S. (2012). Interactive facial feature localization, in: *ECCV*, Springer, 2012, p.
- Leutenegger S., Chli M., Siegwart R.Y. (2011). Brisk: Binary robust invariant scalable keypoints, in: *Proceedings of International Conference on Computer Vision*, Spain, pp. 2548–2555.
- Li N.J., Wang W.J., and Hsu C.C.J. (2015). Hybrid particle swarm optimization incorporating fuzzy reasoning and weighted particle. *Neurocomputing*, 167: pp. 488–501.
- Li X., Ruan Q., Ming Y. (2010). 3D facial expression recognition based on basic geometric features, in: *Proceedings of IEEE 10th International Conference on Signal Processing (ICSP)*, China, pp. 1366–1369.
- Li Y., Mavadati S.M., Mahoor M.H., Ji Q. (2013). A unified probabilistic framework for measuring the intensity of spontaneous facial action units, in: *Proceedings of 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, China, pp. 1–7.
- Liao S., Law M. W. K., and Chung A. C. S. (2009). Dominant local binary patterns for texture classification, *IEEE Trans. Image Process.*, 18(5), pp. 1107–1118.
- Lienhart R., Maydt J. (2002). An extended set of haar-like features for rapid object detection, in: *IEEE ICIP*, IEEE, pp. 900–903.
- Lin J.C., Wu C.H., Wei W.L. (2013). Facial action unit prediction under partial occlusion based on

- error weighted cross-correlation model, in: Proceedings of ICASSP, IEEE, pp. 3482–3486.
- Lisboa, P.J.G. 2002. A Review of Evidence of Health Benefit from Artificial Neural Networks in Medical Intervention. *Neural Networks* 15(1): 11–39.
- Liu C., Wechsler H. H. (2000) . Evolutionary pursuit and its application to face recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (6) 570– 582.
- Liu M., Li S., Shan S., and Chen X. (2015). AU-inspired Deep Networks for Facial Expression Feature Learning, *Neurocomputing*, 159, 2015, pp. 126–136.
- Loderer M., Jarmila P., Oravec M., Mazanec J., (2015). Face parts importance in face and expression recognition. In: International Conference on Systems, Signals, and Image Processing (IWSSIP), pp 188-191.
- Lucey P., Cohn J., Lucey S., Matthews I., Sridharan S., Prkachin K. (2009). Automatically detecting pain using facial actions, in: IEEE International Conference on Affective Computing and Intelligent Interaction , pp. 1–8.
- Lucey P., Cohn J.F., Kanade T., Saragih J., Ambadar Z., Matthews I. (2010). The extended Cohn-Kanade dataset (CK+): A complete expression dataset for action unit and emotion-specified expression, in: Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB'10), San Francisco, USA , pp. 94–101.
- Mahmoodabadi M.J., Mottaghi Z.S., and Bagheri A. (2014). HEPSON: High exploration particle swarm optimization. *Information Sciences* 273: pp. 101–111.
- Majumder A., Behera L., Subramanian V.K. (2014). Emotion recognition from geometric facial features using self-organizing map, *Pattern Recogn.* 47 (3) 1282– 1293.
- Martinez B., Valstar M., Binefa X., Pantic M. (2013) Local evidence aggregation for regression based facial point detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (5) 1149–1163.
- Martinez, Du S. (2012). A model of the perception of facial expressions of emotion by humans: Research overview and perspectives, *J. Mach. Learn. Res.* 13 (1) 1589–1608.
- Matthews I., Baker S. (1988). Active appearance models revisited, *Int. J. Comput. Vis.* 60 (2) (2004) 135–164.
- Meng H. and Bianchi-Berthouze N. (2014). Affective State Level Recognition in Naturalistic Facial and Vocal Expressions, *Cybernetics*, *IEEE Transactions on*, 44(3), pp. 315–328.

- Michie, D et al. 1994. Machine Learning, Neural and Statistical Classification. *Ellis Horwood series in artificial intelligence* 37(4): xiv, 289 .
- Mistry K., Zhang L., Barnden J. (2015). Intelligent facial expression recognition with adaptive feature extraction for humanoid robot, *Neural Networks (IJCNN)* 1-8.
- Mohammed A. W., Zhang M., Johnston M. (2009). Particle swarm optimization based Adaboost for face detection, in: *IEEE Congress on Evolutionary Computation* , pp. 2494–2501.
- Montana G., Parrella F. (2008). Learning to trade with incremental support vector regression experts, 3rd International Workshop on Hybrid Artificial Intelligence Systems (HAIS'08), LNCS, Vol. 5271, Springer, pp. 591–598.
- Moore S., Bowden R. (2011). Local binary patterns for multi-view facial expression recognition, *Comput. Vis. Image Understand.* 115 (4) 541–558.
- Naik M. K., Panda R. (2016). A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition, in: *Applied Soft Computing*, 38 661-675.
- Neoh S.C., Zhang L., Mistry M., Hossain M.A., Lim C.P., Aslam N., and Kinghorn P. (2015). Intelligent Facial Emotion Recognition Using a Layered Encoding Cascade Optimization Model, *Appl Soft Comput.* Volume 34, September, pp. 72–93.
- Niu Y., Shen L., (2006). An adaptive multi-objective particle swarm optimization for color image fusion. *Lecture notes in computer science*, LNCS, pp 473–480.
- Ojala T., Pietikainen M., and Harwood D. (1996). A comparative study of texture measures with classification based on featured distribution, *Pattern Recogn.* 29 (1): pp. 51–59.
- Ojansivu V. and Heikkilä J. (2008). Blur Insensitive Texture Classification Using Local Phase Quantization, *Image and Signal Processing*, Volume 5099, LNCS, pp. 236-243.
- Omran M.G., Salman A.A., Engelbrecht A.P., (2006). Dynamic clustering using particle swarm optimization with application in image segmentation. *Pattern Anal Appl* 2006:332–344.
- Orozco J., Rudovic O., Gonzlez J., Pantic M. (2013). Hierarchical on-line appearance-based tracking for 3d head pose, eyebrows, lips, eyelids and irises, *Image Vis. Comput.* 31 (4) 322–340.
- Ortiz R. (2012). Freak: Fast retina keypoint, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 510–517.
- Pampara G., Franken N., Engelbrecht A.P., (2005). Combining particle swarm optimization with angle modulation to solve binary problems. *IEEE Cong Evol Comput* 1:89–96.

- Pantic M., Valstar M.F., Rademaker R., and Maat L. (2005). Web-based database for facial expression analysis, in Proceedings of IEEE Int'l Conf. Multimedia and Expo, The Netherlands, 317-321.
- Paris S., Kornprobst P., Tumblin J., Durand F. (2008). Bilateral filtering: Theory and applications, Found. Trend Comput. Graph. Vis. 4 (1) 1–73.
- Pendharkar P. C. (2009) Genetic algorithm based neural network approaches for predicting churn in cellular wireless network services, Expert Syst. Appl. 36: 6714-6720.
- Portney, LG, and MP Watkins. 2015. *Foundations of Clinical Research: Applications to Practice*.
- Rajasekhar A, Abraham A, Pant M (2011). Levy mutated artificial bee colony algorithm for global optimization. In: IEEE international conference on systems, man and cybernetics (IEEE SMC 2011), pp 665–662
- Ravindran S. S., Neoh S.-C., Muthusamy H. (2014). A hybrid expert system for automatic detection of voice disorders, Int. J. Med. Eng. Inform. 6 (3) 218–237.
- Rizon M., Karthigayan M., Yaacob S., Nagarajan R., (2007). Japanese face emotion classification using lip features. In: Geometric Modelling and Imaging (GMAI'07), pp 140-144.
- Roweis S., Saul L. (2000). Nonlinear dimensionality reduction by locally linear embedding, Science 290 2323–2326.
- Russell J.A. (2003). Core affect and the psychological construction of emotion, Psychol. Rev. 110 145–172.
- Sagonas C., Tzimiropoulos G., Zafeiriou S., Pantic M. (2013). 300 faces in-the-wild challenge: The first facial landmark localization challenge, in: Proceedings of IEEE International Conference on Computer Vision (ICCV-W'13), in: 300 Faces in-the-Wild Challenge (300-W), Sydney, Australia.
- Sajan K. S., Kumar V., and Tyagi B. (2015). Genetic algorithm based support vector machine for online voltage stability monitoring, International Journal of Electrical power & Energy Systems, 73: 200-208.
- Savran A., Sankur B., Bilge M.T. (2012). Regression-based intensity estimation of facial action units, Image Vis. Comput. 30 (10) 774–784.
- Sayadi M. K., Ramezani R., Ghaffari-Nasab N. (2010). A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems, Int J Ind Eng Comput.1(1):1-10.

- Schmidhuber J. (2015). Deep learning in neural networks: an overview, in: Neural Networks, Volume 61, pp. 85-117.
- Scholköpfung B., Smola A., Müller K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 1299–1319.
- Senechal T., Rapp V., Prevost L. (2011). Facial feature tracking for emotional dynamic analysis, in: 13th International Conference on Advances Concepts for Intelligent Vision Systems (ACIVS), Belgium, Springer, pp. 495–506.
- Setiono, R., Wee Kheng Wee Kheng Leow, and J.M. Zurada. 2002. Extraction of Rules from Artificial Neural Networks for Nonlinear Regression. *IEEE Transactions on Neural Networks* 13(3): 564–77.
- Shah S. K. (2014). A survey of facial expression recognition methods, IOSR J. Eng. 4 (4) 1–5.
- Shan, Gong S., McOwan P.W. (2009). Facial expression recognition based on local binary patterns: A comprehensive study, Image Vis. Comput. 27 803–816.
- Shojaeilangari S., Yau W., and Teoh E. (2014). A Novel phase congruency based descriptor for dynamic facial expression analysis, Pattern Recognition Letters, 49: pp. 55-61.
- Silva A., Neves A., Costa E., (2002). Chasing the swarm: a predator-prey approach to function optimisation. In: Proceedings of the Mendel 2002—8th international conference on soft computing, pp 103–110, Mendel 2002, Brno, Czech Republic.
- Snedecor G.W., and Cochran W.G. (1967). Statistical Methods, 6th Edition, The Iowa State University Press, Ames, Iowa.
- Soleymani M., Asghari-Esfeden S., Fu Y., and Pantic M. (2015). Analysis of EEG signals and facial expressions for continuous emotion detection, IEEE Transactions on Affective Computing.
- Storn R., Price K. (1997). Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces, J Global Optimization;11:341-359.
- Sung J., Kim D. (2007). A background robust active appearance model using active contour technique, Pattern Recogn. 40 (1) 108–120.
- Swets D. L., Weng J. (1996). Using discriminant eigen features for image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) 831–836.
- Tenenbaum J., Silva V., Langford J. (2000). A global geometric framework for nonlinear dimensionality reduction, Science 290 2319–2323.

- Tomasi C., Manduchi R. (1998). Bilateral filtering for gray and color images, in: Proceedings of Sixth International Conference on Computer Vision, IEEE, pp. 839–846.
- Tsalakanidou F., Malassiotis S. (2010). Real-time 2d+3d facial action and expression recognition, *Pattern Recogn.* 43 (5) 1763–1775.
- Tzimiropoulos G., Pantic M. (2014). Gauss-Newton deformable part models for face alignment in-the-wild, in: CVPR, IEEE, pp. 1851–1858.
- Vapnik, Vladimir Naoumovitch. 1998. John Wiley & Sons *Statistical Learning Theory*.
- Verma O.P., Aggarwal D., Patodi T., (2016). Opposition and Dimensional based modified firefly algorithm, *Expert Systems with Applications* 44 168-176.
- Viola P.A., Jones M.J. (2001). Rapid object detection using a boosted cascade of simple features, *Comput. Vis. Pattern Recogn.* 2001 (1) 511–518.
- Vukadinovic D., Pantic M. (2005). Fully automatic facial feature point detection using gabor feature based boosted features, in: Proceedings of the International Conference on Systems, Man and Cybernetics, IEEE, pp. 1692–1698.
- Vural E., Cetin M., Ercil A., Littlewort G., Bartlett M., Movellan J. (2008) Automated drowsiness detection for improved driver safety-comprehensive databases for facial expression analysis, in: International Conference on Automotive Technologies.
- Wang H., Rahnamayan S., Sun H., and Omran M. (2013). Gaussian bare-bones differential evolution, *Cybernetics, IEEE Transactions on*, 43(2), pp. 634–647.
- Wang J., Li T., Ren R. (2010b). A real time idss based on artificial bee colony-support vector machine algorithm. In: 2010 third international workshop on advanced computational intelligence (IWACI), pp 91–96
- Weisstein, Eric W. (2015). Circle-circle intersection. From MathWorld-A Wolfram web resource. <http://mathworld.wolfram.com/Circle-CircleIntersection.html> (ac- cessed Jan 15).
- Weizhong Yan, Weizhong. 2012. Toward Automatic Time-Series Forecasting Using Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems* 23(7): 1028–39.
- Wu C.H., Lin J.C., Wei W.L. (2014). Action unit reconstruction of occluded facial expression, in: Proceedings of ICOT, IEEE, pp. 177–180.
- Wu T., Bartlett M., Movellan J.R. (2010). Facial expression recognition using gabor motion energy filters, in: Proceedings of IEEE Computer Society Conference on Computer Vision and

- Pattern Recognition Workshops (CVPRW'10), IEEE, pp. 42–47.
- Xiao R., Zhao Q., Zhang D., and Shi P. (2011). Facial expression recognition on multiple manifolds, *Pattern Recogn.* 44: pp. 107–116.
- Xie X.F., Zhang W.J., Yang Z.L., (2002b). A dissipative particle swarm optimization. In: *Congress on Evolutionary computation (CEC)*, pp 1456–1461.
- Xie X.F., Zhang W.J., Yang Z.L., (2002a). Adaptive particle swarm optimization on individual level. In: *International conference Signal Processing (ICSP)*, pp 1215–1218.
- Xue B., Zhang M., and Browne W.N. (2013). Particle Swarm Optimization for Feature Selection in Classification: A Multi-Objective Approach, *Cybernetics, IEEE Transactions on*, 43 (6), pp. 1656–1671.
- Yan K., Chen Y., and Zhang D. (2011). Gabor Surface Feature for face recognition, *First Asian Conference on Pattern Recognition (ACPR)*. 288 - 292. Beijing.
- Yang X. S. (2008). *Nature-inspired metaheuristic algorithms*, 1st ed. Frome, UK: Luniver Press .
- Yang X.S., (2010). Firefly algorithm Levy's flight and global optimization, *Research and Development in intelligent systems* 26 209-218.
- Yang X.S., Deb S. (2009). Cuckoo search via Levy flights, in: *Proc. IEEE Conf. of World Congress on Nature & Biologically Inspired Computing*, pp. 210–214.
- Yin H. (2002). ViSOM—a novel method for multivariate data projection and structure visualization, *IEEE Trans. Neural Netw.* 13 237–243.
- Yin H. (2008). On multidimensional scaling and the embedding of self-organising maps, *Neural Netw.* 21 160–169.
- Yu K., Wang Z., Zhuo L., Wang J., Chi Z., Feng D. (2013). Learning realistic facial expressions from web images, *Pattern Recogn.* 46 (8) 2144–2155.
- Zavaschi T.H.H., Britto A.S. Jr., Oliveira L.E.S., and Koerich A.L. (2013). Fusion of feature sets and classifiers for facial expression recognition, *Expert Systems with Applications*, 40(2), pp. 646-655.
- Zeng J., Hu J., Jie J., (2006). Adaptive particle swarm optimization guided by acceleration information. *Proc IEEE/ICCIAS* 1:351–355.
- Zeng Z., Fu Y., Roisman G.I., Wen Z., Hu Y., and Huang T.S. (2006). One-Class Classification for Spontaneous Facial Expression Analysis, in *Proceedings of the 7th international Conference on Automatic Face and Gesture Recognition*, pp. 281-286.

- Zeng Z., Hu Y., Roisman G.I., Wen Z., Fu Y., and Huang T.S. (2007). Audio-Visual Spontaneous Emotion Recognition, in LNCS-Artificial Intelligence for Human Computing, Vol. 4451, pp. 72-90.
- Zeng Z., Pantic M., Roisman G.I., Huang T.S. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (1) 39–58.
- Zeng Z., Zhang H., Zhang R., Zhang Y. (2014). A hybrid feature selection method based on rough conditional mutual information and Naive Bayesian classifier, ISRN Appl. Math. 1–11.
- Zhang B., Gao Y., Zhao S., and Liu J. (2010). Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor. IEEE Transactions on Image Processing, 19(2) , pp. 533–544.
- Zhang H., Zhang Y., Huang T.S. (2013). Pose-robust face recognition via sparse representation, Pattern Recogn. 46 (5) 1511–1521.
- Zhang L. and Tjondronegoro D.W. (2011). Facial expression recognition using facial movement features, IEEE Transactions on Affective Computing. Vol.2, Issue 4, 219 - 229.
- Zhang L., Barnden J.A. (2008). Emma: An automated intelligent actor in e-drama, in: Proceedings of IUI, Spain, pp. 409–412.
- Zhang L., Jiang M., Farid D. and Hossain AM. (2013). Intelligent Facial Emotion Recognition and Semantic-based Topic Detection for a Humoid Robot. Expert Systems with Applications, Vol 40, Issue 13, 5160-5168.
- Zhang L., Mistry K., Hossain A. (2014). Shape and texture based facial action and emotion recognition, in: Proceedings of AAMAS'14. (Demo paper), France
- Zhang L., Mistry K., Jiang M., Neoh S.C., Hossain M.A. (2015). Adaptive facial point detection and emotion recognition for a humanoid robot, Computer Vision and Image Understanding 140 93-114.
- Zhang W., Shan S., Gao W., Chen X., Zhang H. (2005). Local Gabor binary pattern histogram sequence (LGBPHS): a novel non-statistical model for face representation and recognition, in: Proceedings of the 10th IEEE International Conference on Computer Vision, pp. 786–791.
- Zhang Y., Gong D., Hu Y., and Zhang W. (2015). Feature selection algorithm based on bare bones particle swarm optimization. Neurocomputing. 148: pp. 150–157.

- Zhang Y., Zhang L., and Hossain M.A. (2015). Adaptive 3D facial action intensity estimation and emotion recognition, *Expert Syst. Appl.* 42: 1446–1464.
- Zhang Y., Zhang L., Neoh S. C., Mistry K., and Hossain M. A. (2015). Intelligent affect regression for bodily expressions using hybrid particle swarm optimization and adaptive ensembles, *Expert Syst. Appl.* 42: 8678–8697.
- Zhang, G.P. 2000. Neural Networks for Classification: A Survey. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* 30(4): 451–62.
- Zhao B., Guo C.X., Cao Y.J., (2005). A multiagent-based particle swarm optimization approach for optimal reactive power dispatch. *Power systems. IEEE Trans Power Syst* 20(2):1070–1078.
- Zheng Z., Jiong J., Chunjiang D., Liu X., Yang J. (2008). Facial feature localization based on an improved active shape model, *Inform. Sci.* 178 (9) 2215–2223.